GLINK™

G&R

http://www.gar.no/

Glink

for

iOS

User Guide

G&R

Glink for iOS, version 2.5.6

Contents

Contents

Contents

Glink for iOS User Guide   *Gallagher & Robertson*

Contents

Contents

Contents

Warranty

While Gallagher & Robertson AS have made every effort to ensure that this software package should function as described, no warranty of any kind, express or implied, is made by us. Gallagher & Robertson AS shall not be liable for any damages, direct, indirect, special, or consequential arising from a failure of this program to operate in the manner desired by the user. Gallagher & Robertson AS shall not be liable for any damage to data or property that may be caused directly or indirectly by the use of this program. In no event will Gallagher & Robertson AS be liable to you for any damages, including lost profits, savings or other incidental or consequential damages arising from your use or inability to use the program, or for any claim by any other party. Any dispute arising in connection with this warranty will be settled under Norwegian law.

Gallagher & Robertson AS confirm their intention that the software should function as described and will make all reasonable efforts to ensure that errors in the software reported in writing to our office in Norway be corrected in future releases of the software.

Gallagher & Robertson AS grant you as a registered user of a Glink for iOS application a license to use the software and to make those copies you deem necessary for your own security. The package itself remains the property of Gallagher & Robertson AS and we expect you to take all reasonable precautions to ensure that illegal copies are not made by a third party.

Privacy policy

Glink does not collect, store, or transmit information about its users, nor usage of the application. The applications access the camera only when the user activates barcode scanning with the built-in camera. Storage is only accessed when the user imports/ exports a configuration, print, or send debug-trace, scroll back-data, screen-data with e-mail. The applications also access Bluetooth settings and pair with Bluetooth devices like keyboards and barcode scanners.

Introduction

Glink for iPad/iPhone/iPod touch is an iOS-version of Gallagher & Robertson's best-selling terminal emulation program, Glink Professional Edition for Microsoft Windows. Glink for iPad/iPhone runs on Apple iPad, iPhone and Apple Mac devices with Apple M1/M2 chip running Apple's iOS operating system, providing access to a wide range of host systems.

Glink is used to access applications running on Bull, IBM, DEC, UNIX and Linux host systems. Glink has emulations for the IBM 3270 and 5250 terminals, Bull Questar DKU7102, DKU7107 and DKU7211 terminals, Bull VIP7800, VIP7760 and VIP7700 terminals, and DEC VT420, VT320, VT340, VT220 and VT102 terminals. ANSI 3.64 is supported for general communications work.

Glink brings you a high quality and proven emulator on your iPad, iPhone or Mac with M1/M2 chip devices.

Features

- IBM3270, IBM5250, Bull DKU7107, Bull VIP7800, Bull VIP7700, Bull DKU7102, VT420, VT320, VT340, VT220, VT102, ANSI 3.64 terminal emulation, all screen sizes
- TN3270, TN5250, TNVIP, Telnet and G&R/Ggate communication to host
- SSL/TLS support for secure communication
- SSH for direct communication with the SSH Daemon
- SSH tunneling for Telnet protocol
- Multiple concurrent host sessions
- Support for program texts in German, French, Portuguese and Norwegian
- Configurable multiline toolbar with function keys and macros
- Configurable hotspots for function keys, option numbers and URLs
- Macro recording for auto-login and for assignment to toolbar and key
- Pop-up standard keyboard with support of international characters
- Tab/shift-Tab and arrow-keys supported on external Bluetooth keyboard
- Colors can be customized
- Multiple host configurations supported
- Export and import of configurations

Introduction

- Supports Managed App Configuration, which allows users to configure Glink remotely through Mobile Device Management
- Optional password protected configurations
- Optional auto-connect at startup
- Optional use of double-tap as Enter/Transmit
- IBM5250 emulation supports Double Byte Character Sets (DBCS) like Chinese, Japanese and Korean
- Configurable scrollback buffer contains history of your host session
- Print or e-mail host print data
- Print or e-mail terminal emulation content or scrollback buffer content
- Zoom and scroll
- Blinking attribute supported
- Blinking cursor supported
- Two finger swipe gesture to move cursor
- Tap and hold to open http:// or https:// URL in internal or external browser
- Tap and hold to open Mail with e-mail address on screen
- Barcode scanning with built-in camera supported
- Linea Pro and Infinea Tab barcode scanner/magnetic card reader support
- Socket Mobile barcode scanners connected in iOS mode (Bluetooth Accessory Protocol)
- Other Bluetooth barcode scanners like Opticon OPN2002 connected as external keyboard

Support

You can contact Gallagher & Robertson (G&R) technical support by sending an e-mail to support@gar.no or by using our contact form on our website.

Please include product name, version number and information about the device you are running the product on in addition to your question or problem description. It may also be useful for us to have a debug trace of your complete host session up to the problem. You can send an e-mail with the debug-content as an attachment directly from the debug screen.

How to purchase/order a Glink license

Glink for iOS licenses can be purchased online on Apple App Store or in Apple Business Manager (ABM). For large quantities, contact us at gar@gar.no for a quote for private Glink Enterprise applications in ABM. Glink for Android can be purchased on the Google Play Store and G&R online web shop.

Delivery and Installation

If you purchase Glink for iOS on Apple App Store, installation and licensing is handled by the Apples App Store app on your iOS device or in the Apple Business Manager store. You must be logged in to Apple App Store with the same account as you purchased the Glink app with

Setup

If you are installing Glink for iOS for the first time, you have no sessions defined. Tap NEW to define a new session. Choose a session name, set the host address and the other basic options before tapping SAVE in the upper right corner. To configure other options tap SETTINGS, to connect to the configured host tap CONNECT. When in the Glink main screen, tap the menu icon in the upper right corner and then Help to learn more about the product. Help is also available from the menu icon in all settings screens, explaining how to use and set all options.

You will also find a description of all available options in this document.

The Glink emulator screen

Action bar

You will find the Glink action bar at the top row of the Glink screen. Here you will find the status of the current session (*Connected/Disconnected*), the *name* of the current session, a *Disconnect icon* that will disconnect the session if tapped, a *Toolbar toggle icon* that will bring the toolbar up/down when tapped, *a Keyboard toggle* that will bring the on-screen keyboard up/down when tapped, an (optional) *ENTER* button that will send Enter to the host when tapped, an (optional) *Debug icon* that will show the content of the debug buffer when tapped, a *Print icon* that will be displayed if Glink has received print and that will show the received print data if tapped. To the very right, you will find the *Menu icon* that will bring up the main menu when tapped. In **Menu -> Settings -> Screen options** you can configure *Use double tap* to toggle the display of the Action bar. The Action bar can initially be off or on and toggled with a double tap on the screen. This way you can free up more space to the Emulation screen.

Emulation screen
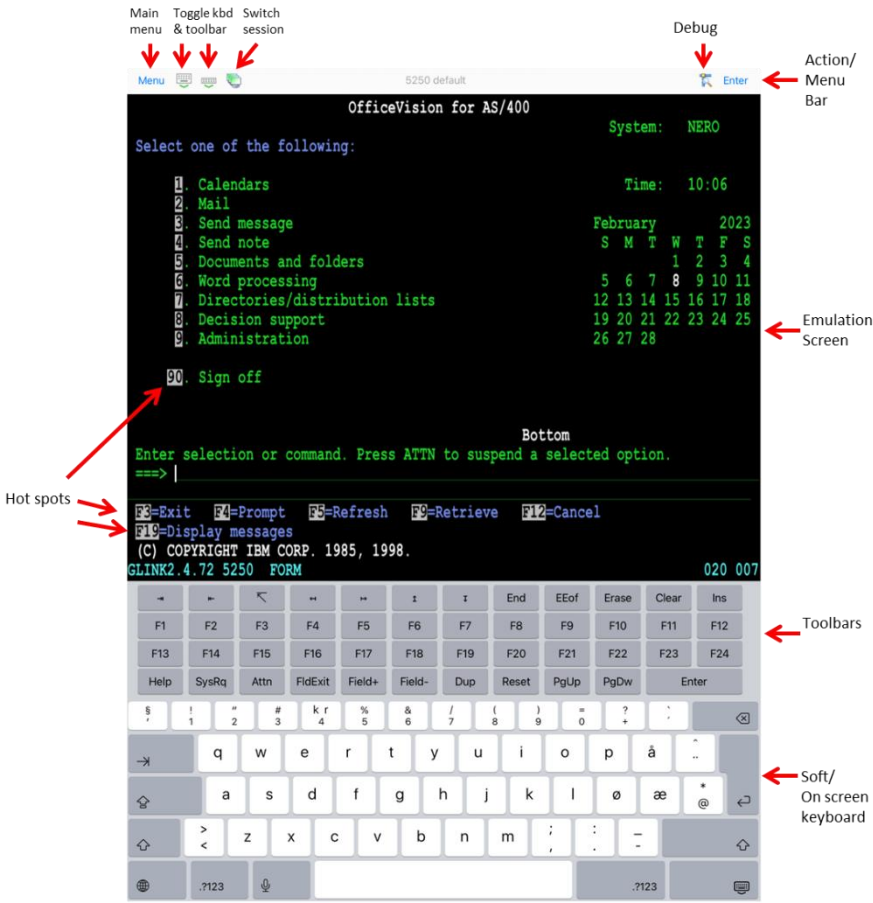
The emulation screen will show the data sent from the host and typed/scanned by the user. In **Menu -> Settings -> Screen options** you can customize the look of the emulation screen. The most important options are the *Font size*, *Show toolbar*, *Limit screen view (x,y)* and the options for *Hotspots*.

The Font size can be set to a specific size between 6 and 36, but it can also be set to Auto or Fit screen. If you have a large display, you probably want to set it to Fit screen. With this setting Glink will select the best font size and stretch/squeeze it to fill the entire screen. If you select Auto, Glink will normally display all rows in portrait mode and all columns in landscape mode, the rest of the host screen can be dragged into view by tap-and-drag left/right/up/down on the screen.

Show toolbar can be set to Auto, Mulitiline, Single-line, Off, Vertical and Horizontal. In Auto mode Glink will switch between Single-line and Multiline, depending on the available space.

Limit screen view (x,y) is very useful if your host application only use the upper left part of the screen, for example 20 columns and 10 rows. Then you would set this parameter to 20,10 and maybe combine it with Font size Fit Screen.

In **Menu -> Settings -> Screen colors** you set the background color, foreground (text) color and you can adjust the colors themselves.

Toolbars

The toolbars are fully configurable. The toolbar is scrollable horizontally and you can have as many rows/bars and buttons as you want. In Multiline mode all rows are shown, while in Single-line mode only one row is shown at the time, but you can toggle between the different toolbars using the toggle-icon on the very left on the toolbar. You can also use the Toolbar toggle icon on the action bar to show/hide the toolbar. In Menu -> Settings -> Toolbar setup you can add, modify, remove and move (remove/paste) buttons and add/remove toolbars (rows) and adjust the button size.

Soft/on-screen keyboard

Glink can use any standard IOS keyboard available on the device. This means that you can switch between keyboards for different languages, default IOS keyboard or other keyboards.

Physical keys and keyboards

Glink supports physical keys on the device itself and external keyboards connected with USB or Bluetooth. All keys can be mapped to terminal functions and macros in Menu -> Settings -> Keyboard ->

Getting started

Create a new session

Glink always enters the *Sessions* screen at startup. Tap the *New configuration* button to define a configuration for your host.

Give your new configuration a name and type the IP address of your host system. The IP address may be a numeric address of the form 129.1.1.150, or a symbolic name (such as www.gar.no). If you need to connect to a port other than the default, this may be added to the end of the address, using a colon (:) to separate it from the rest of the address. For example, to connect to port 1311 you should specify 129.1.1.150:1311 or *www.gar.no:1311*.

If you are using the multi-emulation version of Glink, you can also choose terminal emulation and host communication protocol for the configuration in this screen. Tap *Save* when you are done.

Repeat the process to configure more host systems. For IBM and VT hosts you may now be ready to go, and if you select the session in the sessions screen a connection attempt will be made. For Bull host systems and some IBM hosts you need to con-figure Ggate/TNVIP/TN3270/TN5250 protocol options before you can connect to your host system. Select the configuration and tap *Menu > Settings* to enter directly into the configuration menus. If you try to *Connect* to the selected configuration and the connection fails, then you must go to *Menu > Settings* to correct or complete the connection parameters. This is also the place to go if you need to modify the IP ad-dress of your host system.

Start a session

Tap *Menu > Sessions* to see the list of available configurations. Tap the name of a configuration to select it and connect. Glink displays the main screen and connects to the selected host. If a Connect macro is defined for this session, it will be exe-cuted.

The session menu list is displayed when Glink is started. In the *General options* in the Action bar drop down menu in the main screen you can configure Glink to auto connect to the first session in the list instead.

To run multiple sessions, select *New session* at the bottom of the Sessions screen under the *SESSIONS* STARTED label. If more than one session is configured, select the session to start.

Switching session

If you have started more than one session, the session switch icon (multiple green screens) is displayed at the top bar, next to the keyboard icon. Tap this icon to see a list of active sessions. The current session has a check mark. The icon to the left of the name of the session shows the connection state, green for connected and red for disconnected. Tap a session name to switch to it.

You can also switch session by tapping a name in the Sessions started list in Menu-> Sessions screen.

Zoom and font size

Use two fingers to zoom in and out and one finger to scroll/move the screen. In Menu-> Settings-> Screen options you can change the initial Font size and Font style, Cursor type and configure 3D effect for input fields.

Text size

The default value for a tablet device is to adjust the text size to match the screen size. To do that, the characters are stretched or squeezed.

The default value for a smaller device (phone) is to select a text size (font size) that is readable and, in most cases, suitable for the screen size.

Use zoom or scroll to see hidden parts of the display. The initial text size is set in *Menu->Settings->Screen options->Font size.*

Keyboard

Glink uses the standard on-screen iOS keyboard. The keyboard is activated by tapping the *keyboard* icon next to the *Menu* button or the main screen close to the cursor position or an input field. In the latter case the cursor will be positioned in the tapped field. Tap the *keyboard* icon next to the *Menu* button or main screen outside input fields or on a different line than the cursor is positioned, to remove the keyboard. The keyboard can be configured to always be up in *Menu > General options*.

External keyboard

When using an external keyboard, set the option Menu-> General options-> Keyboard up on. Glink will then be ready for keyboard input when connected to a host application (otherwise you have to tap at an input field first).

Glink support several keyboard shortcuts for function and control keys when using an external Bluetooth keyboard, see *Keyboard shortcuts* for a list.

Device buttons and external keyboard buttons can also be remapped, select Settings->Keyboard and then see *Keyboard mappings* button.

Keyboard shortcuts

The default shortcuts for external keyboard are listed below. Glink also supports user defined key mapping. The listed shortcuts can therefore be remapped.

Shortcuts for keyboards with function keys

For all emulations:
```
F1 to F12          : F1 to F12
```

For IBM emulations:
```
F13 to F24         : shift+F1 to shift+F12 or alt+F1 to alt+F12
```

For VT emulation:
```
F11 to F20         : shift+F1 to shift+F10 or alt+F1 to alt+F10
```

Shortcuts for keyboards without function keys

For IBM 5250, IBM 3270 and VT emulations the following shortcuts are defined:
```
F1  to F10        : Alt+1 to Alt+0
F11 to F20        : Ctrl+1 to Ctrl+0
F21 to F24        : shift+Ctrl+1 to shift+Ctrl+4
```

For DKU or VIP emulations the following shortcuts are defined:

```
F1  to  F10        : Alt+1 to Alt+0
F11 and F12        : shift+Ctrl+1 to shift+Ctrl+2
shift/F1, shift/F10: Ctrl+1 to Ctrl+0
Shift/F11,shift/F12: shift+Ctrl+3 to shift+Ctrl+4
```

Other shortcuts

```
Escape            : Escape
Tab               : Tab
Backtab           : Shift/Tab
Home              : Alt+Cursor up
End field         : Alt+Cursor right
Arrow keys        : Cursor up/down/left/right
Connect           : Alt+C
Disconnect        : Alt+D
Break             : Alt+B
New/next window   : Alt+N
Show this page    : Alt+H
```

IBM 5250 shortcuts

```
Field+            : Alt+  +
Field-            : Alt+  -
FieldExit         : Ctrl+ +
Attn              : Esc
SysRq             : Ctrl+ Esc
PageUp            : Ctrl+Cursor Up
PageDown          : Ctrl+Cursor Down
```

VT shortcuts

```
Find              : Alt+ F
Select            : Alt+ S
Remove            : Alt+ R
PageUp            : Ctrl+Cursor Up
PageDown          : Ctrl+Cursor Down
```

Keyboard mapping

Glink is set up with a default keyboard mapping. You may override the default mapping or add a mapping for a hardware button (key) on the device itself or a key on an external keyboard.

The list box displays the keys mapped if any.

Press the key or button to map (i.e. Volume up button). Select the keyboard icon button to bring up the soft keyboard if need.

Once a key is pressed, a new row is added. Select the **Map** button to map the key to a function or macro.

If the key you want to map is the last one of two input keys, select **No** for mapping the first key and **Yes** for the second one.

If a macro is selected, select the **Macro** button to define the actual macro.

Select the **Remove** option from the drop-down list to remove a key mapping.

Toolbar

The toolbar can be configured to show a single-line toolbar with a *select* button to step through the toolbar lines or as a multiline toolbar to show all the toolbar lines. Even when the multiline toolbar option is selected, Glink will in most cases use the single-line toolbar to save space when the keyboard is up. The default value for this option is to use multiline toolbar for a big screen (for example tablet) and single-line for a small screen (for example phone).

The default toolbars differ from emulation to emulation, but in general there is one for cursor movements, one or two for function keys and one for other keys. All toolbars can be customized to your needs, and you can add macros to all of them with your own text on the buttons. The toolbar area becomes horizontally scrollable if there are more buttons defined than can be displayed on the screen. Go to *Menu > Settings > Toolbar setup* to customize your toolbar, or to *Menu > Settings > Screen options* to change the setting of the *Use multiline toolbar* option.

Hotspots

Glink can be configured to create dynamic push buttons for screen text that refer to function keys, number options and URLs.

For example, if *F1* is part of the screen text, it will be highlighted as a push button and the function F1 will be assigned to it and executed when tapped.

Likewise, if *1=Search* is part of the screen text, the number **1** will be highlighted. When tapped, this number is added to the current cursor position and optionally followed by the *Enter* function to transmit it.

Copy/Paste

Tap and hold to bring up a copy/paste popup menu. Tap and hold a word to select and copy a word, tap and hold the empty space area in front of the word to select and copy the rest of the line. Tap and hold an input field to paste to or copy from this field. To clear the clipboard, tap and hold an area of the screen where the rest of the line is empty.
In the Scrollback buffer, you can select and copy text in the same way as you for example do in the Safari browser.
The paste function may also be added as a button to your toolbar.

Hyperlinks and mail addresses

If URLs or mail addresses are displayed, long-tap a hyperlink or mail address to bring up a popup menu. For URLs you may display the content in Glink or in a browser app. If a mail address is selected, you get the option to send a mail to the addressee.

Export and Import configurations

You can export your current set of configurations for back-up purposes or to share them with other users. The export function is an option in the action bar drop down menu. You will be asked if you want to include Connect macros before you are taken to the Send Mail function with your configurations attached to the message.

The configurations are sent as a file-attachment named config.xxx, where xxx is either glinki, glink5250, glink3270 or glinkvt; depending on the App you are using. These filename-extensions are associated with Glink when you install Glink. This filename-extension is associated with Glink. This means that if you receive a Glink configurations file as an e-mail attachment and export it to the files system, you can import it into Glink by tapping the file. Glink will be started and inform you that you have received a new set of Glink configuration files. If you tap the *Install* button to install it, you will get a new set of configurations that will replace your current configurations. All your current configurations will be overwritten and lost.

The export and import functions allow administrators to prepare and customize configurations for a large group of users and distribute the configurations by e-mail, web-page, dropbox, sky-drive and other utilities that can receive files on the IOS device.

MDM configuration support

If Glink is deployed by a mobile device management (MDM) system, the Glink configuration can be completely controlled by the MDM system, making Glink ready for use without any need for configuration by the end user. See the *Glink MDM configuration parameters* section.

Proxy server

If the network connection to your host is unstable and there is a risk of losing the connection to the host from time to time, you should consider running our *GlinkProxy application* on a server.

The GlinkProxy application runs on a Windows server that can be located near the host computer. It will allow the traffic from the mobile computers with Glink to be routed through the GlinkProxy server to the host. The GlinkProxy server will keep the connection to the host application even if the connection to the mobile computer is lost. When Glink re-connects, the session with the host application will resume as if the connection was not lost.

See the section about *The GlinkProxy Session Persistence Server*.

Gestures

In the main screen Glink recognizes the following gestures:
- *Tap*
  If allowed by the emulation, moves the cursor to the position (normally within an input field)
- *One finger drag*
  Moves the screen content if larger than the screen size
- *Two finger drag*
  Moves the cursor
- *Two finger pinch zoom*
  Zooms the Glink screen
- *Double tap*
  Resets zooming or optional does the send (Enter) function
- *Tap and hold*
  Shows copy/paste options

Menu
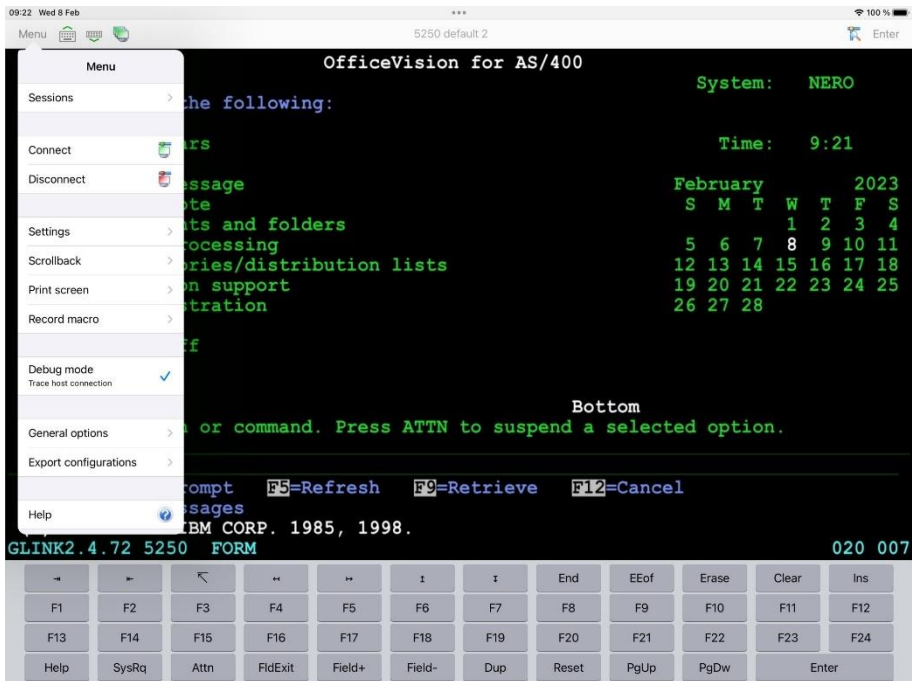
The menu in the main emulator screen has the following choices:

- Sessions
- Connect
- Disconnect
- Settings
- Scrollback
- Print screen
- Record macro
- Debug mode – trace host connection
- Print logging (VT, ANSI, VIP, DKU, 7700, 7102)
- General options
- Export configurations
- Help

Sessions
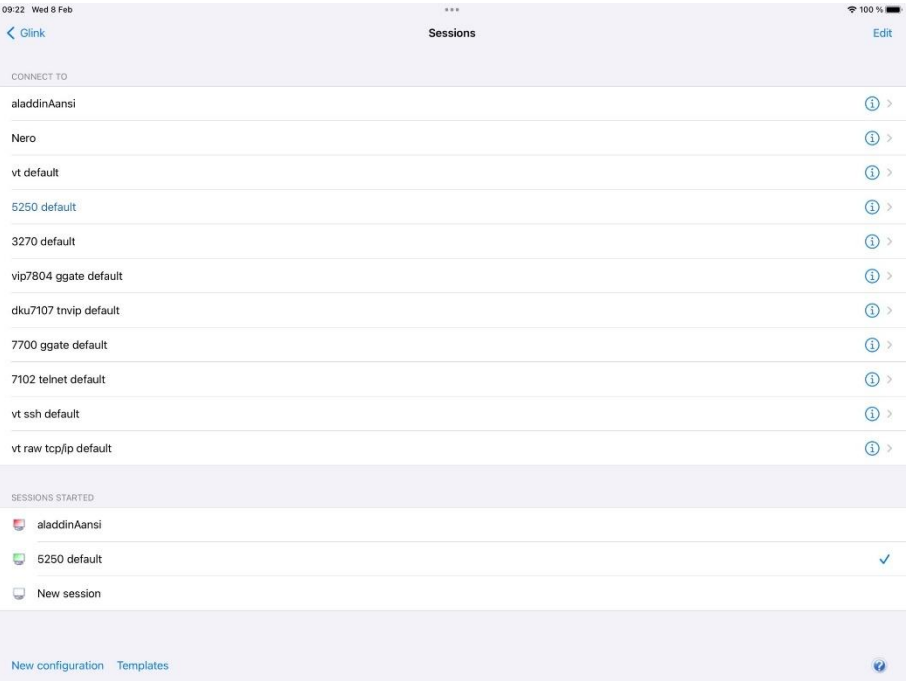
The Sessions screen allows you to change the current host session and add new host configurations. Most parameter settings are per session. The current configuration is displayed in blue. By tapping Settings in the Menu, you can change the configuration parameters for that session. By tapping a different session than the one that has a check mark, you can switch the main screen to this session.

Connect

Tap here to connect to the host specified in the current session configuration. If a Connect macro is defined for this session, it will be executed.

Disconnect

Tap here to disconnect the current session.

Settings

Tap here to modify configuration parameters for the current session. Below you see the Settings screen for a Glink 5250 session.

Scrollback

Tap here to enter the Scrollback buffer. Here you will see all the data that has scrolled off the top of your main screen and all the forms that you have accessed since you connected the current Glink session, provided that the size of your buffer is big enough. You configure the Scrollback length (in pages) in *Menu > Settings > General*. The content of the scrollback buffer can be printed or sent as an e-mail attachment and you can mark and copy selected text. If text is marked when you tap the *Mail* or *Print* button, only the marked text will be sent/printed. Tap the *Glink* button to return to the main screen.

Print screen

Tap here to print or e-mail the current screen content. It can be printed/e-mailed either as text or as a JPEG image. Tap the *Glink* button to return to the emulation window.

Record macro

Tap here to record a macro. Before you tap the *Start recording* button to start recording, you must choose whether you want to record a *Toolbar macro* that you will assign to a button on the toolbar (default) or if you want to record a *Connect macro* that will be executed every time you connect this session to the host, or if you want to record a *Keyboard macro* that you want to assign to a physical key. The *Connect macro* is typically used to logon on to the host, start the desired host application and position the user in the desired startup form. *Toolbar macros* and *Keyboard macros* are typically defined for frequently used operations to avoid time consuming navigation and typing.

When recording a macro, you can use the *Pause* button next to the *Enter* button to specify that you want to wait for user input or that the execution of the macro should pause a number of milliseconds at this stage. This is useful if the host unlocks the keyboard and allows keyboard-/macro-input before it has completed its output to the terminal. User input can be collected at macro startup or at the stage where the input is required.

When you are at the end of the macro function you wanted to record, you go to *Menu > Record Macro* and assign a button text and tap *Save recording* or press a key that you want to map the recorded macro to. The macro will now be saved to the current toolbar or to the key you pressed. You can move it to a different toolbar from *Menu > Settings Toolbar setup*. If you made a typing error during the recording, you must tap *Cancel* in *Menu > Record Macro* to delete the recorded macro and go back to the main screen. *Toolbar macros* are removed when you remove their buttons from the toolbar. *Keyboard macros* are removed when you remove their keyboard mapping from *Menu > Settings > Keyboard mapping*. To remove a *Connect macro* you first must tap *Connect macro* in *Menu > Record Macro*, then tap *Remove connect macro*. If you want to define a completely new *Connect macro*, you now tap the *Start recording* button. If you don't tap *Remove connect macro* before you tap the *Start recording* button, you will extend the current *Connect macro*.

Pause the macro

At any point during the macro recording sequence, you can specify that the execution of the macro can be suspended for either user input or for a time interval.

Wait for user input

Header text
Specify the prompt value.

Ask user at macro start
The user is requested for input either at startup or at the stage where the input is required.

Password
If checked, the input field is set as a password entry field.

Input value
You can also specify an input value that is added to the current screen but not recorded. For example, if a password is requested, your password should not be recorded if recording is to continue.

Wait for a time interval

Specify the number of milliseconds the macro should be suspended at this stage.

Debug mode

Tap here to enable/disable *Debug mode*. In this mode all data sent to and from the host is written to a Debug screen. When enabled, you will see a *Debug* icon at the top right of the screen, next to the *Enter* button. Tap the *Debug* icon to go to the Debug screen. The content of the debug screen can be printed or sent as an e-mail attachment, and you can mark and copy selected text. If text is marked when you tap the *Mail* or *Print* button, only the marked text will be sent/printed. Tap the *Glink* button to return to the main screen.

Debug

When *Debug mode* is enabled all data sent to and from the host is written to this Debug screen, and you will see a *Debug* icon at the top right of the screen, next to the *Enter* button. Tap the *Debug* icon to go to the Debug screen. The content of the debug screen can be printed or sent as an e-mail attachment, and you can mark and

copy selected text. If text is marked when you tap the *Mail* or *Print* button, only the marked text will be sent/printed.

You enable the text-marking feature by holding a finger on the screen until the magnifier appears. When you release the finger, you will have a start- and end-marker that you can move to the desired positions.

Tap *Clear* to clear the debug screen.

Tap *Print* to Print the debug screen. If text is marked when you tap the *Print* button, only the marked text will be printed. If no text is marked, you are allowed to select a Page Range before you print. You must select a printer and how many copies to print before tapping the *Print* button.

Tap *Options* to set Print options, like *Font size* and *Page orientation*.

Tap *Mail* to Mail the debug screen. If text is marked when you tap the *Mail* button, only the marked text will be sent. The debug text is sent as a file-attachment named debug.txt.

Tap *Mail config* to Mail your set of configurations to yourself for back-up purposes or to someone else for sharing. The configurations are sent as a file-attachment named config.xxx, where xxx is either glinki, glink5250, glink3270 or glinkvt; depending on the App you are using. When you tap *Mail config*, you will be asked if you want to include Connect macros (if you have any) before you are taken to the Send Mail function with your configurations attached to the message.

Tap *Arrow-up* icon to position at the start of the Debug buffer.

Tap *Arrow-down* icon to position at the end of the Debug buffer.

Tap the *Glink* button to return to the main screen.

Print logging

Tap here to toggle this option. When enabled a check mark is shown and all sent and received data is logged/captured to the print buffer.

Not available for Glink5250 and Glink3270

General options

These options are common for all the sessions defined.

Never Auto-Lock when connected

When selected, this option disables the IOS "sleep" setting when Glink is the active (visible) app and is connected to the host application.

Toolbar - Click on keypress

Off   No click
Sound 1   Click sound 1
Sound 2   Click sound 2

Show iOS status bar

The iOS status bar is by default always displayed at the top of the screen. If you prefer to use this space for the emulation screen in Glink, you can turn off the iOS status bar so that it is never displayed or only displayed in Portrait mode.

This option has no effect for versions prior to iOS 7.

Keyboard up

Default
Glink will show the keyboard when you click at an input field and hide it when you click outside an input field.

On
This option will always show the keyboard. You can still hide the keyboard with the keyboard button located at the toolbar.

Off
Keyboard appearance will be controlled by the keyboard button only.

Key repeat

When selected, this option enables you to repeat a keystroke multiple times without pressing the key more than once (external/physical keyboard only).

Auto-connect session at startup

Normally Glink displays the Session screen at startup. Use this option if you want to skip the Session screen and instead connect to the first session defined in the Session screen.

Please note that you can reorder the session list by doing a long press at a entry in the session list, and then move the entry by dragging it when a rectangle is shown around the entry.

Password - Protect Settings

This option prevents users from changing or viewing the configurations. The password is required for the functions, *Settings*, *General options*, *New configuration* function and *Export configurations.*

Export configurations

Use this function to export your configurations by mail for back-up purpose or for sharing. The configurations are sent as a file-attachment named config.glinki You will be asked if you want to include Connect macros (if you have any) before you are taken to the Send Mail function with your configurations attached to the message. To import the configurations, send it to your device by mail and select the attachment.

Help

Tap here to enter the online help pages.

Connect icon (Green monitor)

Tap here to connect to the host specified in the current session configuration. If a Connect macro is defined for this session, it will be executed.

Keyboard icon

Tap here to bring the on-screen keyboard up/down.

Toolbar icon

Tap here to bring the on-screen toolbar up/down.

Settings - Communication

TN5250 options

TN5250 is a variation of the Telnet protocol and is used by the IBM5250 emulation to connect to IBM hosts over a TCP/IP network. You can either connect directly to the host if it's running a Telnet server, or via a TN5250 gateway.

Server address

The Server address is normally the TCP/IP address to the server/gateway you connect through to reach the host, for example a G&R Ggate server (using native Ggate

protocol, TNVIP, TN3270, TN5250 or Telnet), a Bull TNVIP server, an IBM TN3270 server, an IBM TN5250 server or the host itself.

The address may be a numeric address of the form 129.1.1.150, or a symbolic name (such as www.gar.no).

If you need to connect to a port other than the default, this may be added to the end of the address, using a colon (:) to separate it from the rest of the address. For example, to connect to port 1311 you should specify 129.1.1.150:1311 or www.gar.no:1311.

The default ports in Glink are:

```
G&R Ggate native protocol    30841
Telnet, TNVIP, TN3270, TN5250      23
```
Note that the Bull TNVIP servers on Unix/Linux or Windows generally use non-standard ports, normally 7323, and you must add this to the IP address.

IPv6 address

If an IPv6 address is specified and you have to specify a port number, use the following syntax:

  [IPv6 address]:port

for example

  [2001:db8:85a3:0:0:8a2e:370:7334]:673

5250 model

This option is provided so that you may select the appropriate 5250 model. This will be used to set the screen size, adjust the screen attribute handling, and to provide terminal identification to the host machine where appropriate. All of the models have screens with 24 rows of 80 columns, except the 3477-x models which have 27 rows of 132 columns:

```
5555-C01 is 24 x 80 DBCS (Double-Byte Char. Set) display
5555-B01 is 24 x 80 DBCS monochrome display
3477-FC  is 27 x 132 display
3477-FG  is 27 x 132 monochrome display
3180-2   is 27 x 132 monochrome display
3179-2   is 24 x 80 display
```

```
3196-A1  is 24 x 80 monochrome display
5292-2   is 24 x 80 display
5291-1   is 24 x 80 monochrome display
5251-11  is 24 x 80 monochrome display
3812-1   is printer
5553-B01 is DBCS printer
```

If you select a printer type, you will probably also need to specify the printer device name to connect to.

Device name

This option allows you to choose a specific device name (DEVNAME) when you connect to a TN5250 server or AS/400.

Some special characters in the device name field can be used to generate the device name:

```
  =         Collision Avoidance ID
  *         Short Session ID
  %         Session Type ID
  &COMPN    Device network host name (Android version)
  &COMPNBT  Device Bluetooth name (Android version)
  &WIP      Wifi IP address (Android version)
  &MAC      Mac address (Android version)
  &MAC_     Short Mac address (Android version)
  &MAC#n    Last n characters of the Mac address (Android
version)
```

The Collision Avoidance ID enables the generation of a new Device name if the Telnet server rejects the previous name. The Collision Avoidance ID has a value in the range 1-9 and A-Z. For example, if ABC= is specified, ABC1, ABC2 and so on will be generated. If ABC== is specified, a random 2-character string will replace the == characters, and so on for ABC=== and ABC====.

The Short Session ID gets the value A if you start one session only. If you start multiple sessions within Glink, the second session will get the Value B and so on. For example, if ABC* is specified, the first session will get ABCA. If that name is already in use, the session will be rejected.

The Session Type ID replaces the character % with the character S for Screen models and P for Printer models. For example, if %ABC* is specified, the first session

will get SABCA if the model is a display type and PABCA if the model is a printer type.

Android only
The **&COMPN** is replaced by the network device name (hostname) if defined for the device. For some Android devices only the Device Bluetooth name is set. To pick up the Device Bluetooth name use &COMPNBT.

Android only
The **&WIP** is replaced by the Wifi IP address formatted as a hex string. For example, if the Wifi IP address of the device is 10.66.1.133, the string 0A420185 will represent the address and will be used as the device name.

Android only
The **&MAC** is replaced by the device Mac address formatted as a hex string. The 6 octets mac address is formatted as a 12-character hex string.

Android only
The **&MAC_** is replaced by the last part of the device Mac address formatted as a hex string. The last 3 octets is formatted as a 6-character hex string.

Android only
The **&MAC#n** (where n is a number from 1 to 9) is replaced by the last n characters of the device Mac address formatted as a hex string.

Keep in mind that the device name should not exceed 10 characters. If the resulting string is longer than 10 characters, Glink will strip excessive characters. Normally at the beginning, but if &WIP is used, Glink will cut up to 4 of the first characters in the IP address before stripping off characters from the beginning of the generated device name. For example, will the first device name *RFT&WIP=* on a device with IP address 10.66.1.133 become RFT4201851.

Keep alive

Keep alive type

There are two keep alive types, **TCP** and **Telnet**. The TCP type is handled by the OS and the Telnet type is handled by Glink.

TCP keep alive

Specifies how often TCP sends keep alive probes. TCP sends keep alive probes to verify that an idle connection is still open. By default, keep alive probes are sent

after 2 hours of inactivity or not at all. It depends on the version of the operating system. The number of keep alive probes that are sent is also system specific.
The keep alive parameter allows you to set both the keep alive time and the keep alive interval in seconds. The format is **time, interval**

For example:

`1200,5`
Specifies that keep alive probes will be sent after 20 minutes of inactivity, and then repeated every 5 seconds until the max number of keep alive probes has been reached.

`1800`
Specifies that keep alive packets will be send after 30 minutes of inactivity, and then repeated every second until the max number of keep alive probes has been reached. One-second interval is the default value.

Telnet keep alive

Specifies how often Glink sends a telnet No Operation Packet (NOP). In this case only an interval can be specified in seconds. The format is **interval**
For example:

`60`
Specifies that a telnet NOP will be sent after one minute of inactivity, and then repeated every minute of inactivity.

Bypass signon using

If the host is configured for *Auto sign-on*, you can specify the *User ID* and *Password* to bypass the initial sign-on screen.

If the host does not support this functionality or is not set up for it, check the *Use alternate auto login* parameter. Glink will instead fill in the *User ID* and *Password* in the Login screen to do an *Auto-login*.

Please note if a connect macro is defined it will be ignored if the *Auto-login* is defined.

Printer options

The AS/400 will require the following device information when connecting a 5250 printer session.

Message queue name

This option tells the AS/400 the name of the message queue to use when sending operational messages. The default AS/400 message queue name of QSYSOPR should be used in most cases. Consult your AS/400 system administrator before changing this option.

Message queue library

This option tells the AS/400 the name of the library containing the message queue used for sending operational messages. The default AS/400 message queue library of *LIB should be used in most cases. Consult your AS/400 system administrator before changing this option.

Font ID

This option tells the AS/400 the font identifier to use when printing.

Form feed type

This option tells the AS/400 the form feed type to use for printing:
- Automatic sheet feeder
- Continuous paper roll form feeder
- Manual sheet feeder

Secure sockets

Use secure sockets - SSL

Check this option to use a secure connection to the host. The host side must of course support this type of connection. The default value for the port number is 992 (GGate 30851). If the host side uses another port number, specify the port number in the host address field, for example:

```
myhost.com:842
```

When connected, a lock icon will be shown in the toolbar. Tap the icon to view the host certificate information.

IPv6 address

If an IPv6 address is specified and you have to specify a port number, use the following syntax:

```
[IPv6 address]:port
```

for example

```
[2001:db8:85a3:0:0:8a2e:370:7334]:673
```

Authenticate server certificate

Glink verifies that the server certificate is from a Trusted root certificate authority in order to accept the SSL connection.

File name client certificate

Specify the name of the client certificate if the server needs to validate the Glink client.

Android version only

If the configured client certificate file is not found, Glink will look for the file in the auto import directory. If still not found, the user gets the option to import a client certificate file using the configured file name.

iOS version only

If the client certificate is not found, the user will get a warning message that the client certificate file must be imported.

See also:
How to import a PSKC #12 certificate (.p12 file)

Password client certificate

Supply the PSKC #12 password if defined.

How to import a private key file

Glink will handle files with *.glinki* as the file name extension, i.e *myfile.glinki*. One way to import it is to send it as a mail to your device with *myfile.glinki* as an attachment. Click on the attachment and Glink will ask if you want to import it as a new configuration (your configuration will not be overwritten in this case).
The *myfile.glinki* must have the following format:

```
[file:my_pk_file]
Add the content of the private key here...
```

Glink will create a file with the file name *my_pk_file* with the content supplied.
In the Telnet or SSH configuration dialog box, enter *my_pk_file* in the *Private key file* field.

TN3270 options

TN3270 is a variation on the Telnet protocol and is used by the IBM3270 emulation to connect to IBM hosts over a TCP/IP network. You can either connect directly to the host, if it's running a Telnet server, or via a TN3270 gateway.

Server address

The Server address is normally the TCP/IP address to the server/gateway you connect through to reach the host, for example a G&R Ggate server (using native Ggate protocol, TNVIP, TN3270, TN5250 or Telnet), a Bull TNVIP server, an IBM TN3270 server, an IBM TN5250 server or the host itself.

The address may be a numeric address of the form 129.1.1.150, or a symbolic name (such as www.gar.no).

If you need to connect to a port other than the default, this may be added to the end of the address, using a colon (:) to separate it from the rest of the address. For example, to connect to port 1311 you should specify 129.1.1.150:1311 or www.gar.no:1311.

The default ports in Glink are:

G&R Ggate native protocol   30841
Telnet, TNVIP, TN3270, TN5250   23
Note that the Bull TNVIP servers on Unix/Linux or Windows generally use non-standard ports, normally 7323, and you must add this to the IP address.

IPv6 address

If an IPv6 address is specified and you have to specify a port number, use the following syntax:

   [IPv6 address]:port

for example

   [2001:db8:85a3:0:0:8a2e:370:7334]:673

3270 model

This option is provided so that you may select the appropriate 3270 model. This will be used to set the screen size and to provide terminal identification to the host machine where appropriate. You may need to select an extended type (one of those with an E extension) to enable use of extended attributes from the host. The base model number determines the alternate screen size the host may use, where

```
327x-1 is 16 rows of 80 columns
327x-2 is 24 rows of 80 columns
327x-3 is 32 rows of 80 columns
327x-4 is 43 rows of 80 columns
327x-5 is 27 rows of 132 columns
3287-1 is printer
```

If you select a printer type, you will probably also need to specify an LU name and select extended telnet mode in the host settings.

LU name

This option allows you to choose a specific LU name. It is only functional when you connect to a TN3270 server that has support for the option and should otherwise be left blank.

Use Extended TN3270

When this option is enabled, the program will use the extended (RFC1647) telnet protocol to talk to the TN3270 server (this will obviously only work if the server supports RFC1647). This protocol allows for more flexible configuration, selection and use of the TN3270 session.

Send Associated LU

This option should be enabled when the specified LU name for a printer (3287-1 type) is the name of a screen device which has been defined with an associated printer on the TN3270 server. The option is only available when you use extended TN3270 to talk to the server, and when an associated printer has been defined for the screen in question.

Keep alive

Keep alive type

There are two keep alive types, **TCP** and **Telnet**. The TCP type is handled by the OS and the Telnet type is handled by Glink.

TCP keep alive

Specifies how often TCP sends keep alive probes. TCP sends keep alive probes to verify that an idle connection is still open. By default, keep alive probes are sent after 2 hours of inactivity or not at all. It depends on the version of the operating system. The number of keep alive probes that are sent is also system specific.
The keep alive parameter allows you to set both the keep alive time and the keep alive interval in seconds. The format is **time, interval**

For example:

```
1200,5
```
Specifies that keep alive probes will be sent after 20 minutes of inactivity, and then repeated every 5 seconds until the max number of keep alive probes has been reached.

```
1800
```
Specifies that keep alive packets will be send after 30 minutes of inactivity, and then repeated every second until the max number of keep alive probes has been reached. One-second interval is the default value.

Telnet keep alive

Specifies how often Glink sends a telnet No Operation Packet (NOP). In this case only an interval can be specified in seconds. The format is **interval**

For example:

```
60
```
Specifies that a telnet NOP will be sent after one minute of inactivity, and then re-peated every minute of inactivity.

Secure sockets

Use secure sockets - SSL

Check this option to use a secure connection to the host. The host side must of course support this type of connection. The default value for the port number is 992 (Ggate 30851). If the host side uses another port number, specify the port number in the host address field, for example:

```
myhost.com:842
```

When connected, a lock icon will be shown in the toolbar. Tap the icon to view the host certificate information.

IPv6 address
If an IPv6 address is specified and you have to specify a port number, use the fol-lowing syntax:

```
[IPv6 address]:port
```

for example

```
[2001:db8:85a3:0:0:8a2e:370:7334]:673
```

Authenticate server certificate

Glink verifies that the server certificate is from a Trusted root certificate authority in order to accept the SSL connection.

File name client certificate

Specify the name of the client certificate if the server needs to validate the Glink client.

Android version only
If the configured client certificate file is not found, Glink will look for the file in the auto import directory. If still not found, the user gets the option to import a client certificate file using the configured file name.

iOS version only
If the client certificate is not found, the user will get a warning message that the client certificate file must be imported.

See also:
How to import a PSKC #12 certificate (.p12 file)

Password client certificate

Supply the PSKC #12 password if defined.

How to import a private key file

Glink will handle files with *.glinki* as the file name extension, i.e *myfile.glinki*. One way to import it is to send it as a mail to your device with *myfile.glinki* as an attachment. Click on the attachment and Glink will ask if you want to import it as a new configuration (your configuration will not be overwritten in this case).
The *myfile.glinki* must have the following format:

```
[file:my_pk_file]
Add the content of the private key here...
```

Glink will create a file with the file name *my_pk_file* with the content supplied.
In the Telnet or SSH configuration dialog box, enter *my_pk_file* in the *Private key file* field.

Auto-login

If both the parameters below is specified, Glink will try to login to the host application.

Please note if a connect macro is defined it will be ignored.

Login with
Specify the login string.

Password
Specify the password.

Note
A *Tab* is performed after the Login string and a *Transmit* is performed after the password. If the parameter includes a function string at the end, the default function will be ignored. The syntax for the *Tab* function is `^"f58"` and for the *Transmit* function is `^"f44"`

The example shows how to do a *Transmit* after the name instead of a *Tab* function.

```
Login with:   MyName^"f44"
Password:     MyPassword
```

External validation

This option allows you to configure an external app for access validation.

In Glink at connect time:

- Start a validation app if not already started
- Send a broadcast to this app to request validation.
- Listen for Broadcast reply from the started app.

Validation app:

- It will start a Broadcast receiver at startup.
- When a request is received, do validation and reply with a positive or negative reply broadcast to Glink.

Settings – Communication

Please contact G&R for more information.

Telnet options

< Settings	Telnet options	ⓘ
Server address		xxx.gar.no
Required		
Terminal type		vt220
Line terminator		CR+LF >
Use IP for BRK		◯
Binary mode		◯
Simulate parity		◯
Keep alive type		TCP >
Keep alive interval		
TELNET LOGIN		
Wait for login prompt		
Login with		
Wait for password prompt		
Password		
SECURE SOCKETS		
Use secure sockets - SSL		◯
Authenticate server certificate		◯
File name client certificate:		
Password client certificate:		

SSH CONNECTION

Use SSH connection

Server address:

User name:

Password:

Private key file:

Keep alive interval in seconds   NB

Server address

The Server address is normally the TCP/IP address to the server/gateway you con-
nect through to reach the host, for example a G&R Ggate server (using native Ggate
protocol, TNVIP, TN3270, TN5250 or Telnet), a Bull TNVIP server, an IBM
TN3270 server, an IBM TN5250 server or the host itself.

The address may be a numeric address of the form 129.1.1.150, or a symbolic name
(such as www.gar.no).

If you need to connect to a port other than the default, this may be added to the end
of the address, using a colon (:) to separate it from the rest of the address. For exam-
ple, to connect to port 1311 you should specify 129.1.1.150:1311 or
www.gar.no:1311.

The default ports in Glink are:

G&R Ggate native protocol   30841
Telnet, TNVIP, TN3270, TN5250   23
Note that the Bull TNVIP servers on Unix/Linux or Windows generally use non-
standard ports, normally 7323, and you must add this to the IP address.

IPv6 address

If an IPv6 address is specified and you must specify a port number, use the follow-
ing syntax:

   [IPv6 address]:port

for example

[2001:db8:85a3:0:0:8a2e:370:7334]:673

Terminal type

This option specifies the string that should be returned by Glink when a Telnet terminal type enquiry is made by the host at connect time.

Line terminator

Normally in ASCII sessions input lines are delimited with CR+LF. Some hosts may demand that you use CR+NUL and others just CR; this option allows you to talk to such hosts.

Use IP (Interrupt Process) command for break

Normally Glink will send a break to the host as a standard Telnet break. Some applications are coded in such a way as to require an Interrupt Process command. If 'break' is not behaving in the way you expect then enable this option to see if this helps.

Binary mode

When this option is set, new sessions will be opened as binary sessions rather than ASCII sessions. This will possibly give 'strange' results that will require $stty$ commands to give acceptable screen output.

Keep alive

Keep alive type

There are two keep alive types, **TCP** and **Telnet**. The TCP type is handled by the OS and the Telnet type is handled by Glink.

TCP keep alive

Specifies how often TCP sends keep alive probes. TCP sends keep alive probes to verify that an idle connection is still open. By default, keep alive probes are sent after 2 hours of inactivity or not at all. It depends on the version of the operating system. The number of keep alive probes that are sent is also system specific.

The keep alive parameter allows you to set both the keep alive time and the keep alive interval in seconds. The format is **time, interval**

For example:

`1200,5`
Specifies that keep alive probes will be sent after 20 minutes of inactivity, and then repeated every 5 seconds until the max number of keep alive probes has been reached.

`1800`
Specifies that keep alive packets will be send after 30 minutes of inactivity, and then repeated every second until the max number of keep alive probes has been reached. One-second interval is the default value.

Telnet keep alive

Specifies how often Glink sends a telnet No Operation Packet (NOP). In this case only an interval can be specified in seconds. The format is **interval**
For example:

`60`
Specifies that a telnet NOP will be sent after one minute of inactivity, and then repeated every minute of inactivity.

Simulate parity

The parameter should normally not be used, but if the host needs 7 bits characters with parity, check this option.

Telnet login

If one or more of the parameters below is specified, Glink will try to login to the host. If the login string or the password is not specified, the login parameters will be asked for at connect time.

Please note if a connect macro is defined it will be ignored.

Wait for login prompt

Instructs Glink to wait for the login prompt string before sending the login string, this may for example be login. Please note that the string is case sensitive.

If no string is specified, Glink will wait for the first string received from host.

Login with

Specify the login string. If not specified, the login string will be asked for at connect time.

Wait for password prompt

Instructs Glink to wait for the password prompt string before sending the password, this may for example be Password. Please note that the string is case sensitive.

If no string is specified, Glink will wait for the next string received from host.

Password

Specify the password. If not specified, the password will be asked for at connect time. If a password is not needed, supply a * to avoid that the password is asked for at connect time.

Note
An *Enter* is performed after the Login string and after the password. If the parameter includes a function string at the end, the default *Enter* function will be ignored. In some cases, a *Tab* is needed after the Login string. The syntax for the Tab function is `^"f58"`

The example shows how to do a auto-login to an AS/400 host using VT emulation

```
Wait for login prompt:      1998
Login with:                 MyName^"f58"
Wait for password prompt:   [
Password:                   MyPassword
```

where [is part of an invisible cursor control sequence.

Wait for command prompt

Instructs Glink to wait for the command prompt string before sending the *Do command* string.

If no string is specified, Glink will wait for the first string received from host.

Do command

Specify the command to run after the login. (Optional)

Secure sockets

Use secure sockets - SSL

Check this option to use a secure connection to the host. The host side must of course support this type of connection. The default value for the port number is 992 (GGate 30851). If the host side uses another port number, specify the port number in the host address field, for example:

```
myhost.com:842
```

When connected, a lock icon will be shown in the toolbar. Tap the icon to view the host certificate information.

IPv6 address
If an IPv6 address is specified and you have to specify a port number, use the following syntax:

```
[IPv6 address]:port
```

for example

```
[2001:db8:85a3:0:0:8a2e:370:7334]:673
```

Authenticate server certificate

Glink verifies that the server certificate is from a Trusted root certificate authority to accept the SSL connection.

File name client certificate

Specify the name of the client certificate if the server needs to validate the Glink client.

Android version only
If the configured client certificate file is not found, Glink will look for the file in the auto import directory. If still not found, the user gets the option to import a client certificate file using the configured file name.

iOS version only
If the client certificate is not found, the user will get a warning message that the client certificate file must be imported.

See also:
How to import a PSKC #12 certificate (.p12 file)

Password client certificate

Supply the PSKC #12 password if defined.

How to import a private key file

Glink will handle files with *.glinki* as the file name extension, i.e *myfile.glinki*. One way to import it is to send it as a mail to your device with *myfile.glinki* as an attachment. Click on the attachment and Glink will ask if you want to import it as a new configuration (your configuration will not be overwritten in this case).
The *myfile.glinki* must have the following format:

```
[file:my_pk_file]
Add the content of the private key here...
```

Glink will create a file with the file name *my_pk_file* with the content supplied.
In the Telnet or SSH configuration dialog box, enter *my_pk_file* in the *Private key file* field.

SSH Connection

Use SSH connection

If this option is enabled, you will be able to configure that the host communication will be routed through the SSH server specified addressed.

Please note that if Use SSL Connection and Use SSH Connection both are selected, SSL will be used.

Server address

Specify the address to the SSH server. The default port number for a SSH server is 22 and will be used if not specified. Another port number may be specified after the host name or IP address, using a colon (:) to separate it from the rest of the address.

IPv6 address
If an IPv6 address is specified and you have to specify a port number, use the following syntax:

```
[IPv6 address]:port
```

for example

```
[2001:db8:85a3:0:0:8a2e:370:7334]:673
```

User name

Specify the SSH user name. If not specified, the user name will be asked for at connect time.

Password

Specify the password (passphrase) for client authentication. If not specified, the password will be asked for at connect time. If a password is not needed, supply a * to avoid that the password is asked for at connect time.

Private key file

If a private key is needed for client authentication, specify the file name to a private key file. If the private key is passphrase protected, supply the passphrase in the Password field.

Keep alive interval

Specify the interval in seconds for the keep alive messages sent to host. Default value is 0, which means no keep alive messages are sent.

See also:
How to import a private key file

SSH options

Server address

Specify the address to the SSH server. The default port number for a SSH server is 22 and will be used if not specified. Another port number may be specified after the host name or IP address, using a colon (:) to separate it from the rest of the address.

User name

Specify the SSH user name. If not specified, the user name will be asked for at connect time.

Password

Specify the password (passphrase) for client authentication. If not specified, the password will be asked for at connect time. If a password is not needed, supply a * to avoid that the password is asked for at connect time.

Private key file

If a private key is needed for client authentication, specify the file name to a private key file. If the private key is passphrase protected, supply the passphrase in the Password field. See also how to import a private key file.

Terminal type

Specify the pseudo terminal, for example vt220. A pseudo terminal is required if you want to log on to the SSH server and run Secure Shell using Raw TCP/IP with no port forwarding.

Number of cols

Specify the number of columns for the pseudo terminal, for example 80.

Number of rows

Specify the number of rows for the pseudo terminal, for example 24.

Keep alive interval

Specify the interval in seconds for the keep alive messages sent to host. Default value is 0, which means no keep alive messages are sent.

Wait for command prompt

Instructs Glink to wait for the command prompt string before sending the *Do command* string.

If no string is specified, Glink will wait for the first string received from host.

Do command

Specify the command to run after the login. (Optional)

Ggate options

Ggate is a gateway product available from G&R for UNIX and Windows platforms. It allows Glink to connect over TCP/IP to Bull and IBM hosts which use the ISO/DSA protocol for network communications.

Server address

The Server address is normally the TCP/IP address to the server/gateway you connect through to reach the host, for example a G&R Ggate server (using native Ggate protocol, TNVIP, TN3270, TN5250 or Telnet), a Bull TNVIP server, an IBM TN3270 server, an IBM TN5250 server or the host itself.

The address may be a numeric address of the form 129.1.1.150, or a symbolic name (such as www.gar.no).

If you need to connect to a port other than the default, this may be added to the end of the address, using a colon (:) to separate it from the rest of the address. For

example, to connect to port 1311 you should specify 129.1.1.150:1311 or www.gar.no:1311.

The default ports in Glink are:

G&R Ggate native protocol   30841
Telnet, TNVIP, TN3270, TN5250   23

Note that the Bull TNVIP servers on Unix/Linux or Windows generally use non-standard ports, normally 7323, and you must add this to the IP address.

IPv6 address

If an IPv6 address is specified and you have to specify a port number, use the following syntax:

   [IPv6 address]:port

for example

   [2001:db8:85a3:0:0:8a2e:370:7334]:673

Alternate server address

The Alternate server address specifies an additional IP address as an alternative to the primary server address. When a connection is to be made to a Ggate gateway, the connection will first be made to the primary address (however, see the Random connect option). If the connection is refused, or after a delay the connection has timed out, then a connection will be attempted to the alternate address. This may be used to provide backup connectivity in the case where one of several Ggate servers may be down, and to provide a form of load balancing.

A more extensive backup and load balancing capability may be achieved by connecting via the G&R Host Links network management product, Gproxy. In this case the primary server address would be the address of Gproxy, and the alternate, if used, would be the address of a backup copy of Gproxy. Gproxy returns the address of the Ggate gateway with the least load of those currently operational, and Glink connects to that address.

The address may be a numeric address of the form 129.1.1.150, or a symbolic name (such as *www.gar.no*).

Settings – Communication

If you need to connect to a port other than the default, this may be added to the end of the address, using a colon (:) to separate it from the rest of the address. For example, to connect to port 1311 you should specify 129.1.1.150:1311 or www.gar.no:1311.

IPv6 address
If an IPv6 address is specified and you must specify a port number, use the following syntax:

```
[IPv6 address]:port
```

for example

```
[2001:db8:85a3:0:0:8a2e:370:7334]:673
```

Random connect

The Random connect option is used in connection with the alternate IP address. It may be checked to force a random choice of either the primary server address or the alternate address as the first address to try for when connecting to the Ggate server. This may be found useful in spreading the total load between multiple Ggate servers.

Secure sockets

Use secure sockets - SSL

Check this option to use a secure connection to the host. The host side must of course support this type of connection. The default value for the port number is 992 (Ggate 30851). If the host side uses another port number, specify the port number in the host address field, for example:

```
myhost.com:842
```

When connected, a lock icon will be shown in the toolbar. Tap the icon to view the host certificate information.

IPv6 address
If an IPv6 address is specified and you have to specify a port number, use the following syntax:

```
[IPv6 address]:port
```

for example

```
[2001:db8:85a3:0:0:8a2e:370:7334]:673
```

Authenticate server certificate

Glink verifies that the server certificate is from a Trusted root certificate authority in order to accept the SSL connection.

File name client certificate

Specify the name of the client certificate if the server needs to validate the Glink client.

Android version only
If the configured client certificate file is not found, Glink will look for the file in the auto import directory. If still not found, the user gets the option to import a client certificate file using the configured file name.

iOS version only
If the client certificate is not found, the user will get a warning message that the client certificate file must be imported.

See also:
How to import a PSKC #12 certificate (.p12 file)

Password client certificate

Supply the PSKC #12 password if defined.

How to import a private key file

Glink will handle files with *.glinki* as the file name extension, i.e *myfile.glinki*. One way to import it is to send it as a mail to your device with *myfile.glinki* as an attachment. Click on the attachment and Glink will ask if you want to import it as a new configuration (your configuration will not be overwritten in this case).
The *myfile.glinki* must have the following format:

```
[file:my_pk_file]
Add the content of the private key here...
```

Glink will create a file with the file name *my_pk_file* with the content supplied.
In the Telnet or SSH configuration dialog box, enter *my_pk_file* in the *Private key file* field.

CoName or Resource

This field is used in two different protocols:

Ggate

Entering a CoName will cause your Ggate session to select a specific CoName (Connection Name) entry in the Ggate DSA configuration file on the Ggate gateway. The CoName entry can can define all configuration parameters that otherwise would have to be entered in the Ggate options, including your terminal device type, and associate a printer session with your terminal device. Your Ggate administrator will supply you with a CoName if appropriate. Note that the Ggate options that follow in the host configuration menu allow you to supply the whole range of DSA and terminal parameters. If you have the required permissions these are used to complete or override the CoName entry. If you do not supply a CoName you must supply enough Ggate options to make the connection. The host application mailbox and the host node will always be needed.

TNVIP

Entering a resource name will select a specific configuration in the TNVIP server. The configuration can define all configuration parameters that otherwise would have to be entered manually, and can also define your terminal device type, and associate a printer session with your terminal device. Your TNVIP administrator will supply you with the name of a resource if appropriate.

Ggate protocol

The Ggate protocol may either be DSA (default) or DIWS. This parameter tells the addressed Ggate which protocol to use for the host-communication. DSA is the preferred protocol. For more information, please check the Host Links Ggate manual or Host Links Line Handler manual.

Remote mailbox name

Specifies the name of the application mailbox for the host subsystem to which you are connecting. This parameter corresponds with the **-DMB** parameter used when connecting using the MainWay Terminal manager. Typical values would be:

GCOS8: **TSS**
GCOS7: **IOF, MML**

Remote Mailbox extension

Specifies the mailbox extension of the application to which you are connecting. This parameter corresponds with the **-EXT** parameter used when connecting using the MainWay Terminal manager. The mailbox extension is only used when connecting to CXI (GCOS8) subsystems and will normally contain the LID (Logical ID). The mailbox extension string can contain up to 4 characters. Login to GCOS8 via CXI requires that the host mode be set to CXI. When you define a LID in TP8, one of the parameters is mailbox extension, usually set to be equal to the LID.

Remote node name

For DSA configuration this parameter specifies the actual DSA session control name for the remote system.

For Ggate configuration this parameter is used to identify a host entry in the Ggate configuration file; this entry is a logical name only which can provide you with one of several different ways to access the same host. The equivalent parameter, which would be provided as a -**SCID** parameter if you were connecting through the Main-Way Terminal Manager (or as the **TSAP** in the OSI world) will be found as one of several OSI/DIWS/DSA parameters in the configuration file on the Ggate server.

Local mailbox name

This parameter specifies the local mailbox name to be used for the connection and is equivalent to the -**LMB** parameter you would specify if you were connecting using the MainWay Terminal Manager.

This can be used as an additional user identification for host security. On GCOS8 systems it may also be used to generate a *LID*, while for IBM (OSF/Janus) connections it can be used to select a specific LU or LU pool.

User ID

Specifies the user ID to be supplied to the host subsystem to which you are connecting and is equivalent to the -**PID** or -**USR** parameter you would specify if you were connecting using the MainWay Terminal manager.

This parameter is usually specified when you connect to GCOS8 and GCOS7 systems but is not used for connections to GCOS6 and IBM systems.

Password

Specifies the password for the host subsystem to which you are connecting and is equivalent to the -**PIC** or -**PW** parameter you would specify if you were connecting using the MainWay Terminal Manager.

This parameter is always specified when you connect to GCOS7 systems, sometimes on GCOS8 systems, but seldom for connections to GCOS6 and IBM systems.

Project

Specifies the project identification to be supplied to the host subsystem to which you are connecting and is equivalent to the -**PJ** parameter you would specify if you were connecting using the MainWay Terminal Manager.

This parameter is sometimes specified when you connect to GCOS7 and GCOS8 systems but is not used for connections to GCOS6 and IBM systems.

Billing

Specifies the billing for the host subsystem to which you are connecting and is equivalent to the -**BA** parameter you would specify if you were connecting via the MainWay Terminal manager.

This parameter is most often used when connecting to GCOS7 systems.

Host mode

This parameter is used to select the target host mode. Through the selection of host mode, a set of internal parameters that control the connection with the remote host application is chosen.

The following host modes are available:

DPS8   This mode is used when communicating with DPS8 TSS and DMIV/TP applications.

CXI   This mode is used when communicating with TP8 and other DPS8 applications that use CXI.

DN   This mode is used when communicating with the NOI on a front-end machine.

DPS7   This mode is always used when communicating with GCOS7 systems.

DPS6   This mode is always used when communicating with GCOS6 systems through SNM.

ROUTER This mode is used with GCOS6 NTM. May also be used to connect to MainWay applications.

UNIX   This mode is always used when communicating with another Unix system. Gmailer and GUFT can make such connections.

IBM   This mode is always used when communicating with IBM hosts through the Janus/OSF gateway (MainWay SNA gateway) in the MainWay.

Terminal mode

This parameter specifies the terminal mode to be used for the connection and is supplied to the host application to indicate what type of presentation should be used for the session.

The following terminal modes are accepted as valid terminal types:

Questar:
> DKU7001, DKU7002, DKU7005, DKU7007, DKU7007D, DKU7102, DKU7105, DKU7107, DKU7107D, DKU7211, DKU7211D, DKU722X, DKU9107

VIP7800:
> HDS5, HDS5T, HDS7, VIP7801, VIP7801C, VIP7801T, VIP7801X, VIP7802, VIP7804, VIP7804V, VIP7805, VIP7805V, VIP7814, VIP8800, VIP9800, TXT7801, TXT7802, TXT2255, TWS2255, TWS2255V

VIP7700:
> VIP7700, VIP7760

IBM3270:
> IBM3270, IBM3278, IBM3279, IBM3287, IBM3270A, IBM3278A, IBM3279A

TTY:
> TTY33, TN300, TTU8126, NVT

PRINTER:
> PRT722X, A2, PRT9220, TTU8124, TTU8126, TTU8221, TTU8223

OTHER:
> DSAIP, G8TP, HL64, HL66, HX0000, BROWSER

If you need to use some other model, and know its hexadecimal identification (see the DNS V4 - Terminal Management Manual, 39 A2 24), then you may configure this using **-TM Hxxxx** as an additional parameter.

See the Host Links Gline documentation for more details about the terminal types recognized by Ggate.

GRTS/LID user string

This parameter is used to define the user record for the host subsystem to which you are connecting. The user record is required when the target GCOS8 site has a "Secure Workstation Environment" or is using the ID-CHECK package developed by Bull Norway for securing networks.

In some situations, the LID is generated by combining the value of this parameter with the local mailbox name. The method being used for generating the LID is determined by the -**GRID** and -**NGRID** parameters in the MainWay configuration.

When communicating with GCOS7 hosts the string !**NEW** may be used, which will force GCOS7 to start the dialogue by presenting the top level menu. Normally it would have tried to reconnect at the level you aborted in the previous host session.

IBM log mode

Defines log mode; used for IBM connections only.

Additional parameters

Additional parameters that apply to the DSA or DIWS line handler may be specified here. See the Host Links Gline documentation for more details.

Keep alive

This option allows you to set a 'keep alive' interval for the gateway. When set to a non-zero value, it tells the program to send dummy packets to the gateway with an interval of the specified number of seconds. The gateway will monitor these and will assume that Glink has terminated abnormally or that the PC has been turned off if these packets stop arriving.

Initial turn

This option tells Glink which side has the initial turn to send after the connection is established.

For most cases use the default value. In some cases, for example with the use of con-
nect macro, the initial turn must be specifically set for Glink to work correctly dur-
ing the connecting phase.

Send commands as data

Normally Ggate will check for and intercept data strings that start with **$*$** and in-
terpret these strings as special service commands. For example, the **$*$dis** will
cause Ggate to disconnect the host session.

If you don't want Ggate to look for these strings, check this parameter and strings
starting with **$*$** will be sent to host as normal data.

TNVIP options

TNVIP is a variation on the Telnet protocol and is used by Bull synchronous emulations to connect to Bull hosts via a TNVIP gateway over a TCP/IP network.

Server address

The Server address is normally the TCP/IP address to the server/gateway you connect through to reach the host, for example a G&R Ggate server (using native Ggate protocol, TNVIP, TN3270, TN5250 or Telnet), a Bull TNVIP server, an IBM TN3270 server, an IBM TN5250 server or the host itself.

The address may be a numeric address of the form 129.1.1.150, or a symbolic name (such as www.gar.no).

If you need to connect to a port other than the default, this may be added to the end of the address, using a colon (:) to separate it from the rest of the address. For example, to connect to port 1311 you should specify 129.1.1.150:1311 or www.gar.no:1311.

The default ports in Glink are:

G&R Ggate native protocol   30841
Telnet, TNVIP, TN3270, TN5250   23
Note that the Bull TNVIP servers on Unix/Linux or Windows generally use non-standard ports, normally 7323, and you must add this to the IP address.

IPv6 address

If an IPv6 address is specified and you have to specify a port number, use the following syntax:

   [IPv6 address]:port

for example

   [2001:db8:85a3:0:0:8a2e:370:7334]:673

Resource name

This option specifies the resource name (mailbox name) that Glink will supply to the TNVIP server when logging in. The resource name will correspond to a particular terminal profile defined in the server. If you do not specify a resource name, then the default profile will be used.

Please note the check box to the right of the input field for the Resource name. If you check this, the user will be asked for the Resource name when connecting to the host. If you also supply a Resource name, it will be suggested to the user.

Terminal type

This option specifies the terminal type that Glink will identify itself with when logging on to the TNVIP server.

Initial turn

This option tells Glink which side has the initial turn to send after the connection is established.

For most cases use the default value. In some cases, for example with the use of connect macro, the initial turn must be specifically set for Glink to work correctly during the connecting phase.

Keep alive

Keep alive type

There are two keep alive types, **TCP** and **Telnet**. The TCP type is handled by the OS and the Telnet type is handled by Glink.

TCP keep alive

Specifies how often TCP sends keep alive probes. TCP sends keep alive probes to verify that an idle connection is still open. By default, keep alive probes are sent after 2 hours of inactivity or not at all. It depends on the version of the operating system. The number of keep alive probes that are sent is also system specific.
The keep alive parameter allows you to set both the keep alive time and the keep alive interval in seconds. The format is **time, interval**

For example:

```
1200,5
```
Specifies that keep alive probes will be sent after 20 minutes of inactivity, and then repeated every 5 seconds until the max number of keep alive probes has been reached.

```
1800
```
Specifies that keep alive packets will be send after 30 minutes of inactivity, and then repeated every second until the max number of keep alive probes has been reached. One-second interval is the default value.

Telnet keep alive

Specifies how often Glink sends a telnet No Operation Packet (NOP). In this case only an interval can be specified in seconds. The format is **interval**

For example:

```
60
```
Specifies that a telnet NOP will be sent after one minute of inactivity, and then re-peated every minute of inactivity.

Secure sockets

Use secure sockets - SSL

Check this option to use a secure connection to the host. The host side must of course support this type of connection. The default value for the port number is 992 (GGate 30851). If the host side uses another port number, specify the port number in the host address field, for example:

```
myhost.com:842
```

When connected, a lock icon will be shown in the toolbar. Tap the icon to view the host certificate information.

IPv6 address
If an IPv6 address is specified and you have to specify a port number, use the following syntax:

```
[IPv6 address]:port
```

for example

```
[2001:db8:85a3:0:0:8a2e:370:7334]:673
```

Authenticate server certificate

Glink verifies that the server certificate is from a Trusted root certificate authority in order to accept the SSL connection.

File name client certificate

Specify the name of the client certificate if the server needs to validate the Glink client.

Android version only
If the configured client certificate file is not found, Glink will look for the file in the auto import directory. If still not found, the user gets the option to import a client certificate file using the configured file name.

iOS version only
If the client certificate is not found, the user will get a warning message that the client certificate file must be imported.

See also:
How to import a PSKC #12 certificate (.p12 file)

Password client certificate

Supply the PSKC #12 password if defined.

How to import a private key file

Glink will handle files with *.glinki* as the file name extension, i.e *myfile.glinki*. One way to import it is to send it as a mail to your device with *myfile.glinki* as an attachment. Click on the attachment and Glink will ask if you want to import it as a new configuration (your configuration will not be overwritten in this case).
The *myfile.glinki* must have the following format:

```
[file:my_pk_file]
Add the content of the private key here...
```

Glink will create a file with the file name *my_pk_file* with the content supplied.
In the Telnet or SSH configuration dialog box, enter *my_pk_file* in the *Private key file* field.

Proxy server setup

If the network connection is unstable and there is a risk of losing the connection to the host, you should consider running our GlinkProxy server. The GlinkProxy server runs on a Windows server that can be located near the host computer and supports large numbers of client sessions.

GlinkProxy is designed to provide persistent host connections for Glink clients where the connection from the client device is unreliable. This may be because the device enters sleep mode or goes out of Wi-Fi range. If the client connection to the proxy server is lost for some reason, the server will retain the connection to the host. This allows the client to reconnect to the server again and continue seamlessly from the point at which the connection was lost.

GlinkProxy parameters

Glink 5250, Glink 3270, Glink VT and Glink for Android and iOS can communicate with TN5250, TN3270, Telnet, TNVIP and Ggate protocols via GlinkProxy to host systems. This is enabled in the **Menu → Settings → Proxy server** menu:

When Glink is configured to use a proxy server, it will send server address and keepalive settings to the proxy server. The proxy server will use these parameters to connect to the host application server and maintain the connection to it. Keepalive settings, user disconnect and connection lost settings from Glink will override settings in GlinkProxy if GlinkProxy is set up to allow Glink to change these parameters. For all communications, the proxy server is a transparent communication gateway between Glink and the host application.

Use server

Must be checked to use the GlinkProxy server

Address

The address of the proxy server (not the host application)

Port

The default port number is 30855

Secure communication

Specifies to encrypt the data flow between the Glink client and the Glink Proxy server.

Proxy keeps connection to host in seconds when

User disconnect
Specifies the number of seconds the proxy server should keep the connection to the host application after a user initiated disconnect. The default value is 0 which means that the connection to the host is disconnected immediately. This is what happens without a proxy server.

Connection lost
Specifies the number of seconds the proxy server should keep the connection to the host application after the connection to Glink is dropped, for example due to bad Wi-Fi signals. When the network connection is reestablished, the user can then reconnect and resume the session with the host application.

Settings – Emulation

IBM 5250 emulation options
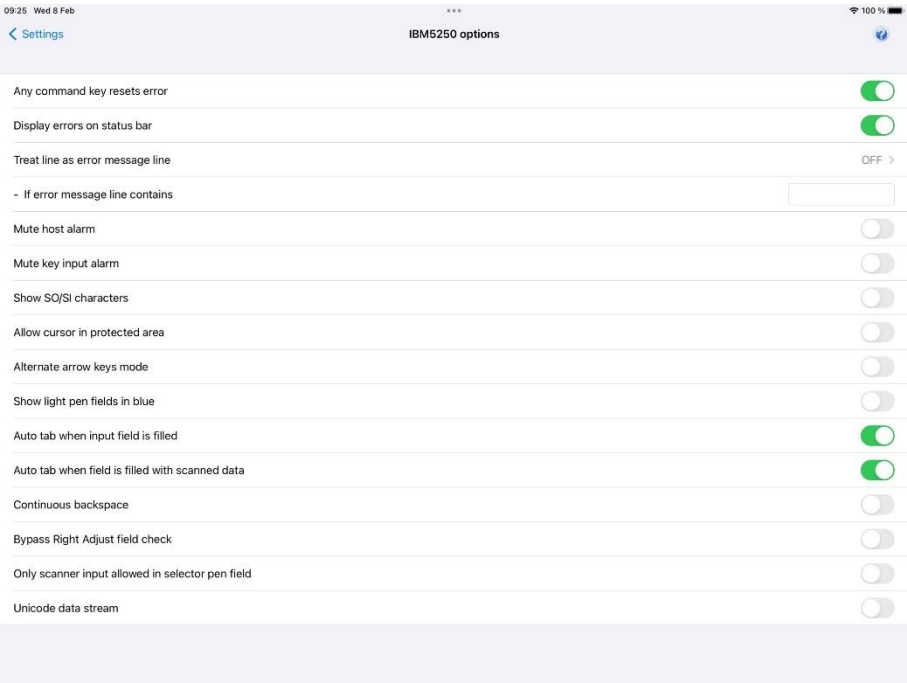
Any command key resets error

If an error has occurred, normally the keyboard is unlocked with the Reset command key only. If this option is checked, any command key will reset the error state and unlock the keyboard.

Display error on status bar

Normally error messages are displayed in line 25 of the 5250 screens. If the option is checked, errors are instead displayed in the status bar of Glink.

Treat line as error message line

Text from the host at the line specified will be regarded as an error text and will be displayed on the status line together with a notification sound. Text starting after position 10 on this line will not be regarded as an error text.

If error message line contains

If one or more search strings are specified, the *error message line* specified above must contain one of these strings to be treated as an error message line.

The whole line is search for a matching string. An optional column parameter can be added, then a matching string must start in that column.

The error message line is display on the status line but can be suppressed with an optional parameter. The notification sound is played in both cases.

The parameter supports the following syntax:

```
text1:text2;col=n,msg=false
```

Please note that search strings are separated with colon. Any optional parameters are preceded with a semicolon. These parameters again are separated with a comma. Some examples:

`MAIN`	The line must contain *MAIN*
`MAIN;msg=false`	The line must contain *MAIN*. No message, only notification sound
`MAIN;col=2,msg=false`	The line must contain *MAIN* at column 2. No message, only notification sound

MAIN:OFCTSK;col=2	The line must contain *MAIN* or *OFCTSK* at column 2.

Mute host alarm

Ignore alarms issued by the host application (The volume control and alarm sound is configured in the iOS/Android Settings).

Mute key input alarm

Ignore key input error alarms (The volume control and alarm sound is configured in the iOS/Android Settings).

Show SO/SI characters

If the host is using double byte character set like Chinese, Japanese or Korean, it might be convenient to see where SO and SI characters are in the input fields.

Allow cursor in protected area

Normally if you point outside an input field, the cursor is not moved to this location. Instead, the keyboard, if displayed, is hidden (removed). If this option is checked, the cursor is instead moved to the protected location without changing the keyboard state.

Alternate arrow keys mode

By default, the arrow keys can move the cursor to any position on the screen. The new position can be in a protected area.

If this option is checked and the arrow key brings the cursor to a protected area, Glink will do a Tab to the next input field for Arrow Down and Arrow Right or do a Back Tab to the previous field for Arrow Up and Arrow Left.

Show light pen fields in blue

Sometimes in can be convenient to distinguish fields that the host application has set as light pen fields from other type of fields.

Auto tab when input field is filled

The cursor is moved to the next field automatically when the current input field is filled. Sometimes it may be convenient to turn off this behavior, for example when input data comes from a scanner with a Tab character as suffix.

Continuous backspace

By default, backspace stops deleting characters at the beginning of the current input field. If this option is checked, backspace will continue to delete characters in the previous input field.

If the device has an alphanumeric keypad that contains both numbers and letters on the same keys, this option will allow input until proper character has been typed at the end of a field without moving to the next field. The reason is that alphanumeric keypad uses backspace when switching character (for the character c, the keyboard will send the characters a+backspace+b+backspace+c

Bypass Right Adjust field check

Input fields that are defined as Right Adjust should be right adjusted with FieldExit, Field+ or Field- and give an error if not.

Some emulators bypass this check and send the field without right adjusting. With this option checked, Glink will do the same.

5250 Transliteration - Host code page
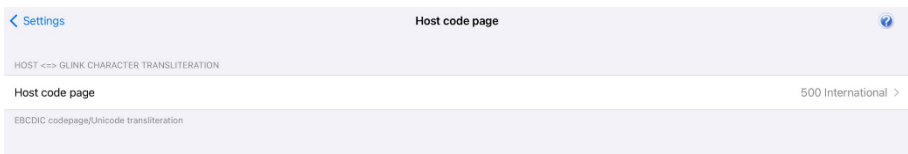
The IBM host is set up to use an EBCDIC code page. Glink needs to know which code page is used by the host to convert EBCDIC characters to Unicode characters used internally by Glink.

Settings – Emulation

Code page	Euro	Countries
037	1140	Australia, Brazil, Canada, New Zealand, Portugal, South Africa, USA
273	1141	Austria, Germany
1388		China
880		Cyrillic
1025		Cyrillic
277	1142	Denmark, Norway
278	1143	Finland, Sweden
280	1144	Italy
930		Japan
930E		Japan (930 Extended)
939		Japan
1399		Japan
933		Korea
1364		Korea
870		Latin 2
284	1145	Latin America, Spain
285	1146	Ireland, United Kingdom
297	1147	France
500	1148	International
	1149	Iceland
1026	1155	Turkey
937		Taiwan
1371		Taiwan

5250 General Options

‹ Settings	**General**	🔵?

Auto tab when input field is filled	🟢
Auto Tab host	⚪
Destructive BS	🟢
Auto scroll to cursor if not visible	🟢

AVAILABLE

Scrollback length (in pages)	32

Auto Tab when input field is filled

If this option is selected, then an automatic tab to the next field of a form will be performed as soon as the last character of the field has been typed. If not selected, then no action will be performed until either the Tab key or the next character is entered.

Auto Tab host

This option controls whether automatic tabbing (move to the next field of the form when the last character of a field is entered) should also be applied to input coming from the host machine. The normal Glink action here is not to perform such automatic tabbing for host data even when local auto-tab is set. The option may be required for host applications developed using emulators that assume that auto-tabbing applies equally to local and host data.

Destructive backspace

Effective only for modes where characters are being echoed locally by the emulator, this option specifies that the backspace key should replace the character backspaced over with a space character. Note that in insert mode backspace will in any case delete such characters and move that part of the line following the cursor one character to the left.

Auto scroll to cursor if not visible

This option controls whether Glink automatically should scroll the screen so that the cursor position is visible on the screen or not. It is by default enabled, and when enabled, Glink will always scroll the screen to display the cursor position. When disabled, Glink will display the top left portion of the screen when it receives a new screen from the server and the display will not follow the cursor when it is moved. The user can scroll the display to the cursor position by tapping the screen or scroll freely by dragging the screen, but each new screen will default to the top left corner.

Scrollback length

This option lets you specify the maximum number of scrollback pages saved. Note that each extra page of scrollback you configure will cost extra memory allocated.

IBM 3270 emulation options

09:39 Wed 8 Feb	⚹ 100 % ▬
‹ Settings	IBM3270 options	ⓘ

Numeric checking	None ›
Treat line as error message line	OFF ›
Mute host alarm	⚪
Mute key input alarm	⚪
Allow cursor in protected area	⚪
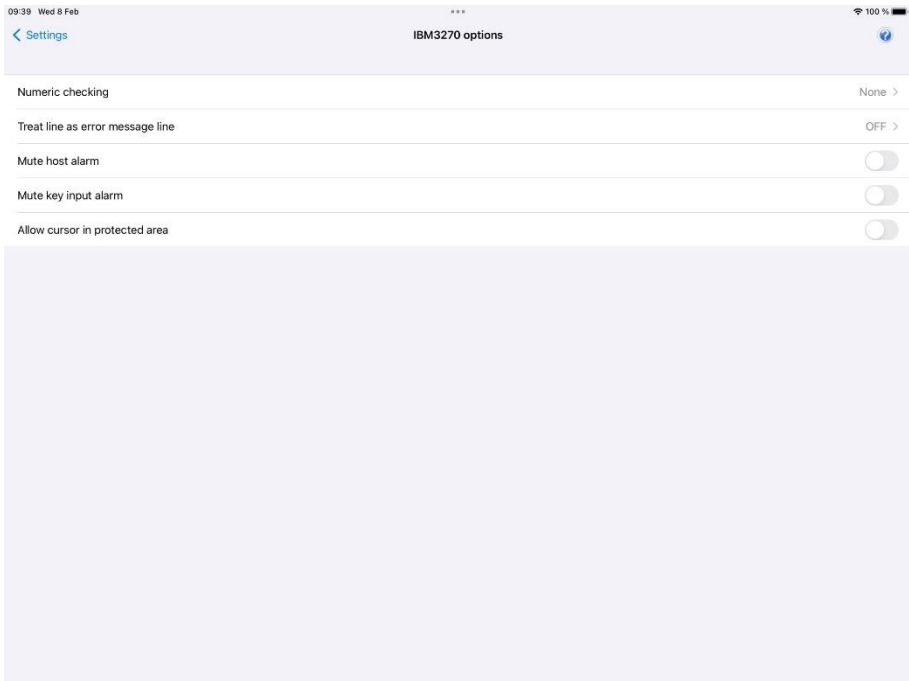
Numeric check

This option specifies what type of numeric checking should be done by the emulator when the host defines an unprotected field as being numeric. There are four possible values that can be set:

`None`

No numeric checking is done. This is equivalent to a Type-writer keyboard selection in some emulators.

`Strict`

Numeric checking is enforced, and only numeric digits, decimal point, comma, and minus sign may be entered in the unprotected field.

Settings – Emulation

Emula-tor	Same as above but also allows plus sign (this is the numeric checking that is implemented in several other emulators).
Re-laxed	Allows all the above, also uppercase letters and shifted numerics (for example "!" and "#"). This is equivalent to a Data Entry keyboard selection in some other emulators.

The default setting is None, i.e., numeric checking is not performed.

Treat line as error message line

Text from the host at the line specified will be regarded as an error text and will be displayed at the status line together with a notification sound. Text starting after position 10 at this line will not be regarded as an error text.

Mute host alarm

Ignore alarms issued by the host application (The volume control and alarm sound is configured in the iOS/Android Settings).

Mute key input alarm

Ignore key input error alarms (The volume control and alarm sound is configured in the iOS/Android Settings).

Allow cursor in protected area

Normally if you point outside an input field, the cursor is not moved to this location. Instead, the keyboard, if displayed, is hidden (removed). If this option is checked, the cursor is instead moved to the protected location without changing the keyboard state.

3270 Transliteration – Host code page
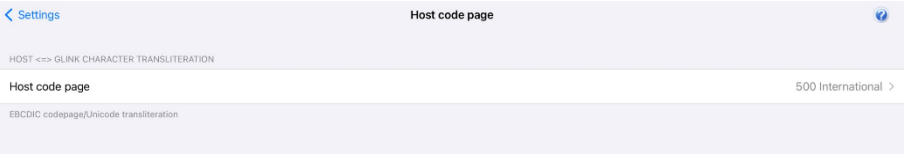
‹ Settings	Host code page	
HOST <=> GLINK CHARACTER TRANSLITERATION		
Host code page		500 International ›
EBCDIC codepage/Unicode transliteration		

The IBM host is set up to use an EBCDIC code page. Glink needs to know which code page is used by the host to convert EBCDIC characters to Unicode characters used internally by Glink.

Code page	Euro	Countries
037	1140	Australia, Brazil, Canada, New Zealand, Portugal, South Africa, USA
273	1141	Austria, Germany
1025		Cyrillic
277	1142	Denmark, Norway
278	1143	Finland, Sweden
280	1144	Italy
870		Latin 2
284	1145	Latin America, Spain
285	1146	Ireland, United Kingdom
297	1147	France
500	1148	International
	1149	Iceland

3270 General Options

Auto Tab when input field is filled

If this option is selected, then an automatic tab to the next field of a form will be performed as soon as the last character of the field has been typed. If not selected, then no action will be performed until either the Tab key or the next character is entered.

Auto Tab host

This option controls whether automatic tabbing (move to the next field of the form when the last character of a field is entered) should also be applied to input coming from the host machine. The normal Glink action here is not to perform such automatic tabbing for host data even when local auto-tab is set. The option may be required for host applications developed using emulators that assume that auto-tabbing applies equally to local and host data.

Destructive backspace

Effective only for modes where characters are being echoed locally by the emulator, this option specifies that the backspace key should replace the character backspaced over with a space character. Note that in insert mode backspace will in any case delete such characters and move that part of the line following the cursor one character to the left.

Auto scroll to cursor if not visible

This option controls whether Glink automatically should scroll the screen so that the cursor position is visible on the screen or not. It is by default enabled, and when enabled, Glink will always scroll the screen to display the cursor position. When disabled, Glink will display the top left portion of the screen when it receives a new screen from the server and the display will not follow the cursor when it is moved. The user can scroll the display to the cursor position by tapping the screen or scroll freely by dragging the screen, but each new screen will default to the top left corner.

Scrollback length

This option lets you specify the maximum number of scrollback pages saved. Note that each extra page of scrollback you configure will cost extra memory allocated.

VT / ANSI options
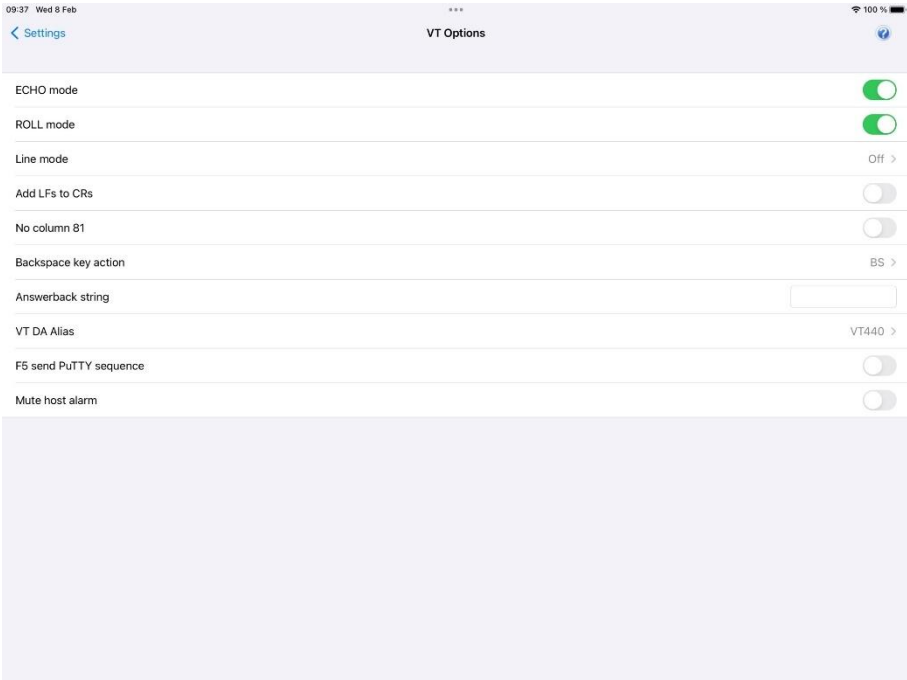
ECHO mode

Echo mode is a mode in which characters you type are not displayed locally, but just sent to the host machine. Usually, the host will send these back to you ('echo' them) as you type. If you are not seeing what you are typing, then possibly you need to turn off Echo mode (in which case Glink will display the characters locally). If you see every character, you type twice, on the other hand, that's a sure sign that Echo mode needs to be turned on.

ROLL mode

Roll mode is the normal method of operation. When a new line is received and the emulator is already positioned on the last line of the screen, all lines of the screen are moved up one position. The original top line of the screen is lost (but will be saved into your scrollback buffer if you have configured one). Reception of a new line in

the bottom line of the screen is an error and will result in the DATA OVERFLOW error message.

Line mode

When a character is typed, it is first sent to the host application, then returned (echoed) back from the host application and finally displayed on the screen. The host application has full control of what is displayed on the screen. This is the default behavior for VT and ANSI emulations.

However, some host applications expect to receive the characters line by line instead (Line mode). In this mode the characters are displayed on the screen when typed, but the characters are not sent to the host until the *Enter* key has triggered a *Line mode* send. In fact, most control keys will trigger a Line send to the host application.

Line mode options

Off	Default, Line mode is not used
Basic	Line mode is on, every control character except the *Backspace* key will trigger a line mode send
Extended	Line mode is on, every control character except the *Backspace*, *Delete*, *Home*, *End*, *Cursor forward* and *Cursor back* keys will trigger a Line mode send.

Please note that the *Echo mode* option should be checked even when *Line mode* is selected.

In Extended Line mode the listed keys will perform a local operation on the typed characters not yet sent. In the VT emulation, the *Home* function is on the *Find* toolbar button and *Home* key, the End function is on the *Select* toolbar button and *End* key, the *Delete* function is on the *Remove* toolbar button and *Delete* key.

Add LFs to CRs

If this option is turned on, then GLINK will automatically add a 'line feed' character to all 'carriage return' characters received from the host. This option may be used in cases where all the data from the host is written on the same line of your screen.

No column 81

This option decides exactly how the emulator treats the right-hand column of the screen. This option should only be needed in exceptional circumstances.

Backspace key action

The backspace key can either send to the host BS (0x08), DEL (0x7f) or BS plus DEL (0x08+0x7f). Normally the Backspace key sends hex 08 character to the host.

Answerback string

This option allows you to configure an 'answerback' string to be used when the host machine sends an ENQuiry character to the emulator. Any string may be entered here. If control characters are required inside the answerback string use the '^' prefix notation, for example ^M for CR.

Mac address (Android version only)
To insert the mac address of the device, use the following key word i n the answerback string:

&MAC Adds the 6 octets mac address as a 12-character hex string

&MAC_ Short mac address, adds the last three octets of the mac address as a 6-character hex string

VT DA Alias

Specifies the Device Attributes (DA) response.

F5 send PuTTY sequence

Sends the PuTTY sequence, ESC OT (1B 4F 54), for the F5 function. The default sequence for F5 function is ESC [15~ (1B 5B 31 35 7E)

VT / ANSI - Transliteration
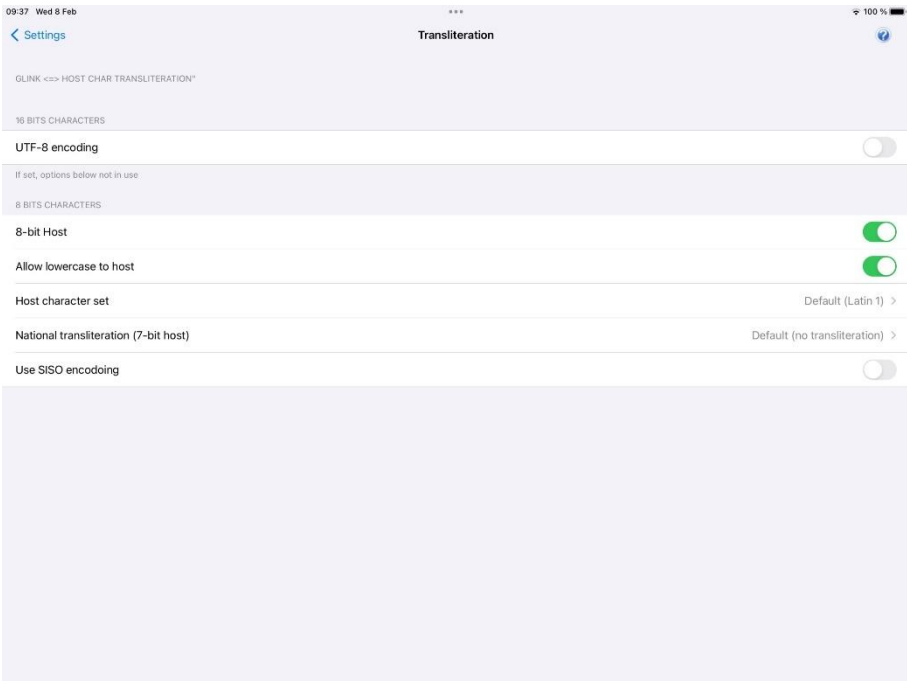
The following options deals with translation/encoding of characters to and from the host. Internally Glink uses Unicode. Most host applications use an 8-bit character set.

UTF-8 encoding

Set this option if the host uses UTF-8 encoding to send/receive Unicode characters. Glink uses Unicode internally, so no further transliterations are needed, and the transliteration options become redundant.

Eight bit host

Most host application uses an 8-bit character set, and this option is on by default.

Allow lowercase to host

When this option is set, lower case data from the keyboard will be sent to the host as is. If the option is unchecked, then any lower-case data will be converted to upper case before being sent to the host.

Host character set

Normally the host is set up to use an 8-bit character set. Glink needs to know which character set is used by the host to convert 8-bit characters to Unicode characters used internally by Glink. The default character set is ISO 8859-1, which is normally used in Western Europe and USA.

National transliteration (7-bit host)

Some host supports 7-bit character set only. The transliteration tables listed are used to convert from 7-bit national characters to 8-bit characters. Then Glink converts the 8-bit character to Unicode.

Use SI/SO for printing

Normally the SI/SO control characters are used to indicate line graphics from the host. Some special-purpose applications use them for print control, however, and for such applications this option must be enabled.

VT / ANSI – General

Settings	General

Destructive BS	🟢

AVAILABLE

Capture on CR	LF ›
Scrollback length (in pages)	32
Initial screen width	80 ›
Initial screen height (rows)	24

Destructive backspace

Effective only for modes where characters are being echoed locally by the emulator, this option specifies that the backspace key should replace the character backspaced over with a space character. Note that in insert mode backspace will in any case delete such characters and move that part of the line following the cursor one character to the left.

Capture on CR (Capture delimiter)

This option decides how CR and LF characters should be treated when capturing or logging to print. When set to CR or LF then a new line will be started when that character is received, and the other will be ignored. When set to CRLF, CR and LF characters will be passed transparently to the capture file (or printer). Normally this option should be set to LF but for certain special applications you may need to

change the option to get correct output on the capture file. This option is not acted upon in VT100 modes, which support data capture compatibly with the built in print functionality on the actual terminals.

Scrollback length

This option lets you specify the maximum number of scrollback pages saved. Note that each extra page of scrollback you configure will cost extra memory allocated.

Initial screen width

This option lets you choose between screen widths of 40, 80 and 132 columns (normally the choice of emulation mode will set this correctly for you).

Initial screen height (rows)

This option lets you choose the number of rows. The default value is 24 rows.

This option is only relevant for VIP7800 and VT emulation modes.

VIP7800 options

Settings – Emulation

The following options are available in Menu > Settings > VIP options when the emulation mode is set to VIP7801/VIP7804.

Auto tab when input field is filled

The cursor is moved to the next field automatically when the current input field is filled. Sometimes it may be convenient to turn off this behavior, for example when input data comes from a scanner with a Tab character as suffix.

Auto Tab host

This option controls whether automatic tabbing (move to the next field of the form when the last character of a field is entered) should also be applied to input coming from the host machine. The normal Glink action here is not to perform such automatic tabbing for host data even when local auto-tab is set. The option may be required for host applications developed using emulators that assume that auto-tabbing applies equally to local and host data. This option applies to VIP and DKU only.

Start in TEXT mode

This option determines whether the emulator should start up in character or text mode. The host applications normally set the mode required.

Extended status

This option enables 'extended status' mode where the emulator will send additional bytes when a remote enquiry is received.

Non-linear forms

This is a special option that tells the VIP emulation mode to use fields in the order they are defined rather than in the order they appear on the screen. Don't use this unless you have an application that has this need, in that the application must be written specifically for this type of form (the main use of this option is for defining forms that allow the user to tab through variable fields in an application-specified order).

Preserve Roll mode

This option tells the VIP emulation that the roll mode setting should be preserved while forms are being shown on the screen. A standard terminal resets roll mode when entering forms mode and will not be in roll mode when going back to text mode unless this is set specifically by the host application. Setting this option will allow you to stay in roll mode when returning to text mode.

Suppress status line

This option tells GLINK not to display data in the status line other than when instructed to do so by the host machine.

TX on return

This option only has an effect when the emulator is started in Text mode. If set on, then the CR key may be used to transmit messages to the host. In addition to transmitting the data, the key will also perform its normal CR function, with an added LF if the Auto LF option is on.

Start in Roll mode

This specifies whether the emulator should initialize itself in roll mode, 'scrolling' the screen when receiving data past the end of the screen. This parameter will often be set for you by the host machine when you log on.

Block mode

This option determines whether the terminal should use block mode when transmitting messages to the host or not. It only has an effect if the emulator is started in text mode and will normally be set by the host at connect time. DO NOT turn on this option unless you are sure you need it - you will otherwise experience apparent 'hangs' when transmitting large blocks of data to the host in text or forms mode.

Disallow status line lock

This option stops the application you are using from locking the status line. Messages presented in the status line will be left there but removed on the first keypress you use after the message is shown.

Space Suppression

This determines whether trailing spaces at the end of lines and/or fields should be removed before the data is sent to the host. This option should normally be turned on.

Add CRLF to transmit

This determines whether messages sent when Space Suppression is not active should have CRLF delimiters sent between each line of the message or not.

Add LFs to CRs

When it's selected, Glink will append a linefeed character to each incoming CR character; this should be used in situations where you see data coming from the host being written repeatedly in the same line of the screen. The opposite also applies; if output from the host is displayed with double spacing this may well be because this option has been selected when it should not have been.

Send NL (New line) after XMT

When this option is set, then a new line operation will be performed when the transmit key is pressed. This is the standard mode, but it may be convenient to suppress it when working with some editors that generate a new line themselves and which would otherwise force you to work with double-spaced lines.

Send ETX for end of transmission

This determines whether messages should be delimited with an ETX or EOT character (this applies to all messages in Forms or Text mode, and to status messages in Character mode).

Answer ENQ with

This specifies the character string that the emulator will identify itself with when answering an enquiry from the host.

7200 attributes

This option specifies whether the emulator should accept the 7200 set high intensity and set low intensity commands. This should not be necessary unless you are using an application written specifically for the 7200 terminals. Note that you may not mix 7200-type and 7800-type attributes on the same screen under any circumstances. An additional setting, 'Extended', allows use of two extra commands for invisible and blinking fields that were supplied with some 7200 hardware emulations.

VIP Transliteration

UTF-8 encoding

Set this option if the host uses UTF-8 encoding to send/receive Unicode characters. Glink uses Unicode internally, so no further transliterations are needed, and the transliteration options become redundant.

Eight bit host

Most host application uses an 8-bit character set, and this option is on by default.

Allow lowercase to host

When this option is set, lower case data from the keyboard will be sent to the host as is. If the option is unchecked, then any lower-case data will be converted to upper case before being sent to the host.

Host character set

Normally the host is set up to use an 8-bit character set. Glink needs to know which character set is used by the host to convert 8-bit characters to Unicode characters used internally by Glink. The default character set is ISO 8859-1, which is normally used in Western Europe and USA.

National transliteration (7-bit host)

Some host supports 7-bit character set only. The transliteration tables listed are used to convert from 7-bit national characters to 8-bit characters. Then Glink converts the 8-bit character to Unicode.

Use SI/SO for printing

Normally the SI/SO control characters are used to indicate line graphics from the host. Some special-purpose applications use them for print control, however, and for such applications this option must be enabled.

VIP General

Destructive BS	🟢

AVAILABLE

Capture on CR	LF ›
Scrollback length (in pages)	32
Initial screen width	80 ›
Initial screen height (rows)	24

Destructive backspace

Effective only for modes where characters are being echoed locally by the emulator, this option specifies that the backspace key should replace the character backspaced over with a space character. Note that in insert mode backspace will in any case delete such characters and move that part of the line following the cursor one character to the left.

Capture on CR (Capture delimiter)

This option decides how CR and LF characters should be treated when capturing or logging to print. When set to CR or LF then a new line will be started when that character is received, and the other will be ignored. When set to CRLF, CR and LF characters will be passed transparently to the capture file (or printer). Normally this option should be set to LF but for certain special applications you may need to

change the option to get correct output on the capture file. This option is not acted upon in VT100 modes, which support data capture compatibly with the built in print functionality on the actual terminals.

Scrollback length

This option lets you specify the maximum number of scrollback pages saved. Note that each extra page of scrollback you configure will cost extra memory allocated.

Initial screen width

This option lets you choose between screen widths of 40, 80 and 132 columns (normally the choice of emulation mode will set this correctly for you).

Initial screen height (rows)

This option lets you choose the number of rows. The default value is 24 rows.

This option is only relevant for VIP7800 and VT emulation modes.

DKU options
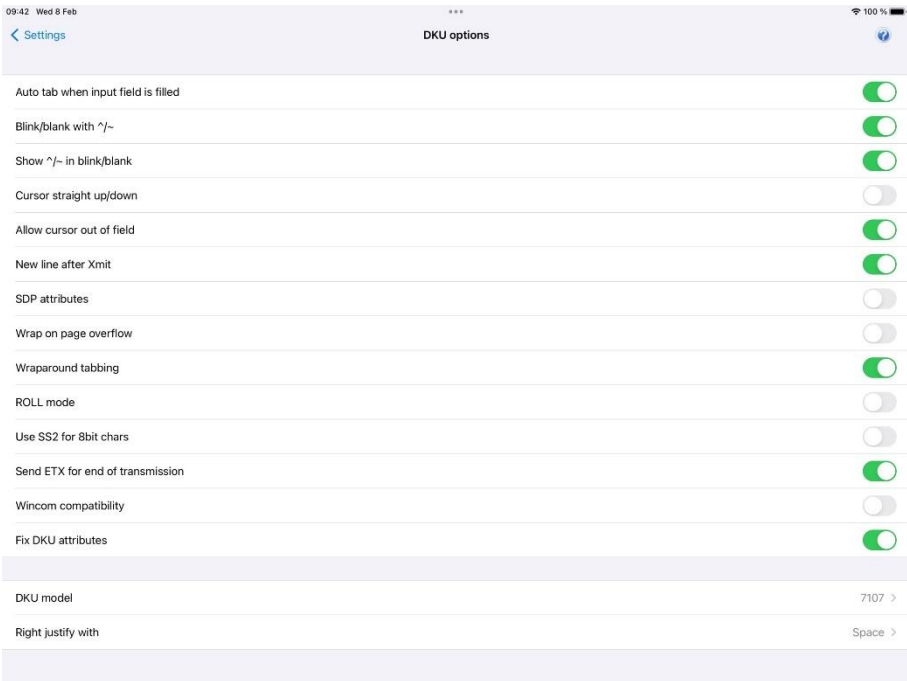
The following options are available from Menu > Settings > DKU options when the emulation mode is set to DKU7107.

Auto tab when input field is filled

The cursor is moved to the next field automatically when the current input field is filled. Sometimes it may be convenient to turn off this behavior, for example when input data comes from a scanner with a Tab character as suffix.

Blink/blank with ^/~

This option tells the emulator that the blink and blank attributes are to be set on reception of the caret (^) and tilde (~) characters respectively.

Show ^/~ in blink/blank

Specifies whether the caret (^) and tilde (~) characters should be shown on-screen when using them to indicate blink/blank screen attributes.

Cursor straight up/down

Setting this option will allow you to use the arrow up/down keys to move to the nearest field in the previous/next lines rather than to the leftmost or rightmost field.

Allow cursor out of field

Setting this option will allow you to move the cursor forwards and backwards out of variable fields when typing into a form.

SDP attributes

SDP attribute rendition, normally set by the host if needed. In this mode, attributes do not take space on the screen, and they are character rather than field oriented.

Wrap on page overflow

Handle host data overflow by wrapping around to the top of the screen rather than waiting for the user to press the Enter key.

Wrap around tabbing

When this option is set (the default) use of tab, back tab, cursor up and down keys will wrap around from the end of the form to the beginning and vice versa. If the option is not set, then use of these keys at the start or end of the form will produce an error message and no cursor movement.

Start in Roll mode

This specifies whether the emulator should initialize itself in roll mode, 'scrolling' the screen when receiving data past the end of the screen. This parameter will often be set for you by the host machine when you log on.

Use SS2 for 8bit chars

If this option is turned ON, Glink will accept sending and receiving extended characters such as accented letters, euros etc. as SS2 encoded sequences.

Send ETX for end of transmission

This determines whether messages should be delimited with an ETX or EOT character (this applies to all messages in Forms or Text mode, and to status messages in Character mode).

Wincom compatibility

This option affects the way input in VIP numeric fields (defined with GS4 rather than with DAQ) is checked. For compatibility with Wincom only the characters + , - . / and 0-9 are allowed. If the option is not checked then normal VIP numeric checking is applied and all characters in columns 2 and 3 of the ASCII table are allowed.

Fix DKU attributes

Some applications try to mark variable fields with attributes to highlight them on the screen. If this is done on the DKU emulation running in non-SDP attribute mode, then the attribute must be defined and reset outside of the actual variable field, and this results in extra marked characters outside of the variable. This option suppresses definition of attributes that overlap variables, for a cleaner look to the emulation screen.

DKU model

Allows you to select the Questar model type that the emulator will return in the reply to a device attribute enquiry.

Right justify with

Allows you to specify which character should be used when doing right justification of fields in the form.

VIP 7700 options

09:43 Wed 8 Feb	🛜 100 %
❮ Settings	VIP7700 options

Auto tab when input field is filled	⬤
Auto Tab host	◯
Start in TEXT mode	⬤
Wraparound tabbing	⬤
Add LFs to CRs	◯
Space suppression	⬤
Suppress status line	◯
ROLL mode	◯
Show 7700 spaces	◯
VIP compatibility	Normal ❯
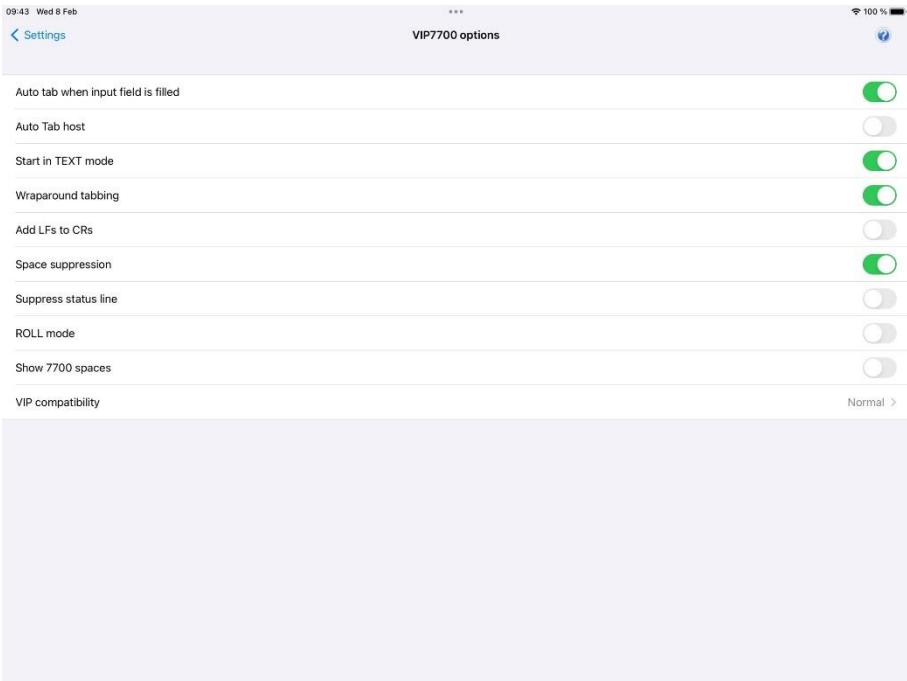
The following options are available in Menu > Settings > 7700 options when the emulation mode is set to VIP7700.

Start in TEXT mode

This option determines whether the emulator should start up in character or text mode. The host applications normally set the mode required.

Wrap around tabbing

When this option is set (the default) use of tab, backtab, cursor up and down keys will wrap around from the end of the form to the beginning and vice versa. If the option is not set then use of these keys at the start or end of the form will produce an error message and no cursor movement.

Add LFs to CRs

When it's selected, Glink will append a linefeed character to each incoming CR character; this should be used in situations where you see data coming from the host being written repeatedly in the same line of the screen. The opposite also applies; if output from the host is displayed with double spacing this may well be because this option has been selected when it should not have been.

Space Suppression

This determines whether trailing spaces at the end of lines and/or fields should be removed before the data is sent to the host. This option should normally be turned on.

Suppress status line

This option tells GLINK not to display data in the status line other than when instructed to do so by the host machine.

Roll mode

This specifies whether the emulator should initialize itself in roll mode, 'scrolling' the screen when receiving data past the end of the screen. This parameter will often be set for you by the host machine when you log on.

Roll mode is the normal method of operation. When a new line is received and the emulator is already positioned on the last line of the screen, all lines of the screen are moved up one position. The original top line of the screen is lost (but will be saved into your scrollback buffer if you have configured one). Reception of a new line in the bottom line of the screen is an error and will result in the DATA OVERFLOW error message.

Show spaces (7700)

Spaces inside VIP7700 variable fields can be marked with a small dot to show their presence. If you prefer such spaces to be shown then check this option.

VIP compatibility

This option controls the compatibility mode in which the 7760 emulation is run. If your application was written specifically for the ITT courier and/or a Thomas Box then you will need to select the appropriate option in order for your application to run correctly.

VIP 7700 Transliteration

UTF-8 encoding

Set this option if the host uses UTF-8 encoding to send/receive Unicode characters. Glink uses Unicode internally, so no further transliterations are needed, and the transliteration options become redundant.

Eight bit host

Most host application uses an 8-bit character set, and this option is on by default.

Allow lowercase to host

When this option is set, lower case data from the keyboard will be sent to the host as is. If the option is unchecked, then any lower-case data will be converted to upper case before being sent to the host.

Host character set

Normally the host is set up to use an 8-bit character set. Glink needs to know which character set is used by the host to convert 8-bit characters to Unicode characters used internally by Glink. The default character set is ISO 8859-1, which is normally used in Western Europe and USA.

National transliteration (7-bit host)

Some host supports 7-bit character set only. The transliteration tables listed are used to convert from 7-bit national characters to 8-bit characters. Then Glink converts the 8-bit character to Unicode.

Use SI/SO for printing

Normally the SI/SO control characters are used to indicate line graphics from the host. Some special-purpose applications use them for print control, however, and for such applications this option must be enabled.

VIP 7700 General

Destructive backspace

Effective only for modes where characters are being echoed locally by the emulator, this option specifies that the backspace key should replace the character backspaced over with a space character. Note that in insert mode backspace will in any case delete such characters and move that part of the line following the cursor one character to the left.

Capture on CR (Capture delimiter)

This option decides how CR and LF characters should be treated when capturing or logging to print. When set to CR or LF then a new line will be started when that character is received, and the other will be ignored. When set to CRLF, CR and LF characters will be passed transparently to the capture file (or printer). Normally this option should be set to LF but for certain special applications you may need to change the option to get correct output on the capture file. This option is not acted

upon in VT100 modes, which support data capture compatibly with the built in print functionality on the actual terminals.

Scrollback length

This option lets you specify the maximum number of scrollback pages saved. Note that each extra page of scrollback you configure will cost extra memory allocated.

Initial screen width

This option lets you choose between screen widths of 40, 80 and 132 columns (normally the choice of emulation mode will set this correctly for you).

Initial screen height (rows)

This option lets you choose the number of rows. The default value is 24 rows.

This option is only relevant for VIP7800 and VT emulation modes.

Settings – Screen

Toolbar setup
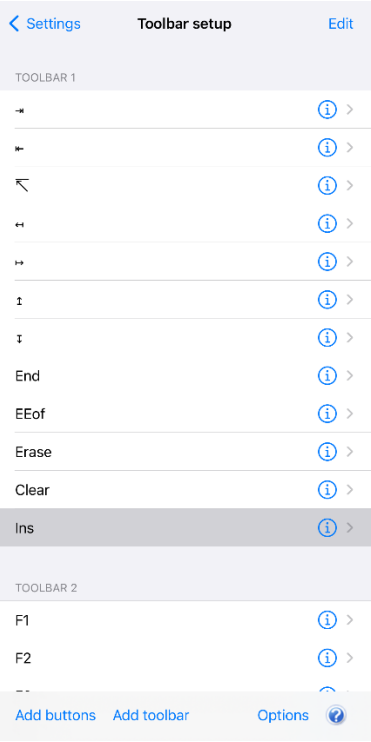
For the *Screen options* -> *Show toolbar* values: *Auto, Multiline* and *Single-line* the toolbars will be displayed horizontally with *Toolbar 1* as the first toolbar row, *Toolbar 2* as the second toolbar row and so on.

For the value *Vertical*, the toolbar is shown on the right side of the screen. The number of toolbars configured will decide the number of toolbar columns, but the toolbar buttons will be distributed from left to right from the top and down, so that the Toolbar 1 buttons will be shown on the first rows, then the Toolbar 2 buttons will follow on the next rows and so on.

For example, like this:

→	←	↖	↩
↦	↕	⟂	End
EEof	Erase	Clear	Ins
F1	F2	F3	F4
F5	F6	F7	F8
F9	F10	F11	F12
F13	F14	F15	F16
F17	F18	F19	F20
F21	F22	F23	F24
Help	SysRq	Attn	FldExit
Field+	Field-	Dup	Reset
PgUp	PgDw	Enter	

For the value *Horizontal*, the toolbar is shown at the bottom of the screen using horizontal page scrolling. The number of toolbar buttons that is displayed on each "page" is decided by the options *Max visual buttons per* row and *Max visual toolbar* rows. The number of toolbar rows shown on the screen can be less than the number of toolbar rows defined. The redundant rows are appended to the end of the visual rows.

Edit

The Edit button allows you to either delete buttons from the toolbars or to move a button to a new position, either within the same toolbar or to a different toolbar. Tap the stop icon and confirm by tapping the Delete button to delete a button. Tap, hold and drag the list-icon on the button you want to move to its new position. Tap Done to exit Edit-mode.

Add buttons

This option allows you to add new buttons to the current position in the current toolbar, so before tapping Add buttons you have to select the toolbar and toolbar-position where you want to insert the new toolbar button(s). From the Add buttons screen you can select one or more buttons to be inserted. The button can either be used to activate a predefined function (key) or a Macro. Tap Save when you have completed your selection or tap Cancel to return to the Toolbar setup screen. If you decided to add a Macro, tap the detail-icon on the righthand side of the new button
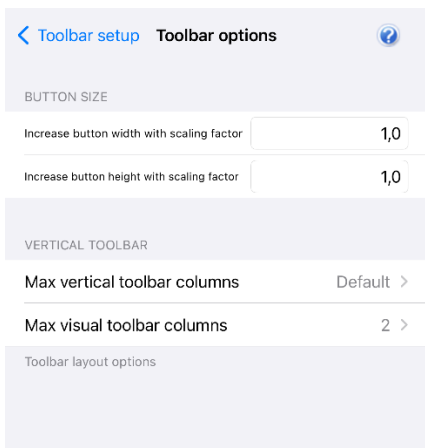
with text Macro to configure the macro to be executed. See Macro string section be-
low on how to define a macro string.

Note that the simplest way to define a Toolbar macro is to use the macro-recording
feature available from Menu > Record macro. See the help page for Record macro

Add toolbar

This option allows you to add new buttons to a new toolbar. From the Add toolbar
screen you can select one or more buttons to be inserted. Tap Save when you have
completed your selection or tap Cancel to return to the Toolbar setup screen.

Options

The size of the toolbar buttons be changed, both the width and the height.

Toolbar options

Increase button width with scaling factor
The original button width can be adjusted with a scaling factor. A factor of 2.0 will
double the size. Try a factor of 1.2 to increase the width slightly.

Please note that the width for an individual button can be doubled by preceding the
button text with a * (star). For example, if the button text is modified from *En-
ter* to **Enter*, the button text shown is *Enter* but the button width is doubled. If the
button text starts with ****Enter**, the button width will be tripled and so on.

Increase button height with scaling factor
The original button height can be adjusted with a scaling factor. A factor of 2.0 will double the size. Try a factor of 1.2 to increase the height slightly.

If a parameter format error is detected, the size is set to the original size.

Toolbar 1-n buttons

Each button in a toolbar can be modified by tapping the detail-icon on the righthand side. You can modify this button-position to perform a different function by tapping the Button selected line. You can also modify the text displayed on the button for this particular function by editing the Button text field. If the Button selected is a Macro, you have to define the Macro string to be executed when the button is tapped.

Button size

This option allows you to adjust the width and height for all the toolbar buttons with a scaling factor (from 0.2 to 5.0).

Please note that the width for an individual button can be doubled by preceding the button text with a * (star). If the button text starts with **, the button width will be tripled and so on. For example, if the button text is modified from Enter to *Enter, the button text shown is Enter but the button width is doubled.

Vertical toolbar

The toolbar is shown on the right side of the screen when *Settings->Screen options->Show toolbar->Vertical* is selected. By default, the number of toolbars columns shown on the right-side screen will match the number of toolbars defined.
The toolbar buttons will be distributed from left to right from the top and down, so that the Toolbar 1 buttons will start on the first row and continue on the next row(s), then the Toolbar 2 buttons will follow on the next rows and so on.

Toolbar layout options

The number of vertical toolbar columns can be configured independently of the number of toolbars defined, see *Settings->Toolbar setup->Options -> Vertical toolbar*. And the number toolbar columns shown on the right side of the screen can

be less than the vertical toolbar columns number enabling horizontal toolbar scrolling.

Max vertical toolbar columns

The default value is the same as the number of toolbars defined. This parameter redefines the number of vertical toolbar columns.

Max visual toolbar columns

The number of columns shown is by default equal to the number of Vertical toolbar columns. If the visual number is less, horizonal toolbar scrolling shows hidden columns. If the number is larger, the default value is used.

Horizontal toolbar layout

The toolbar is shown at the bottom of the screen using horizontal page scrolling. The Horizontal toolbar options are used when *Settings->Screen options->Show toolbar->Horizontal* is selected. The number of toolbar buttons that is displayed on each "page" is decided by the options Max visual buttons per row and Max visual toolbar rows. The number of toolbar rows shown on the screen can be less than the number of toolbar rows defined

When *Max visual buttons per* row is less than the number om buttons configured on a row and you swipe the toolbar horizontally, it will snap to the next "page" of buttons.

Max visual buttons per row

The default value is the number of buttons to fit the screen width.

Max visual toolbar rows

The number of rows shown is by default equal the number of toolbar rows defined. If the number is less, the redundant rows are appended to the end of the visual rows. If the number is larger, the default value is used.

Macro string

When entering a macro string, in addition to plain text, there are several conventions, all of which are signaled using the caret (^). If you wish to enter a 'real' caret, then you must type it twice (^^).

Control characters may be entered using the normal convention with a letter following the caret symbol. For example, a return is ^M and a line feed is ^J.

You may also send specific ASCII codes in hexadecimal, decimal, or octal form by following the caret with a $, #, or & character, and the desired code:

`^#ddd` decimal specification
`^&ooo` octal specification
`^$hh`   hexadecimal specification

Note that decimal and octal codes must be 3 digits and prefixed with 0 if it is 2-digit codes.

Transmitting the string you have defined on a macro can be awkward if you are working in multiple environments where you sometimes need a CR terminator and sometimes need an ETX or EOT. To do this transparently, use the conventional form ^! (caret + exclamation mark). The correct line terminator will be sent depending on the mode in which the emulator is operating. Note that if a macro contains multiple transmits then the macro execution will be suspended on each transmit and resumed when the host has responded. This functionality is only reliable for host connections with a 'turn' mechanism that signals when the host response is finished (DSA, Ggate and TNVIP).

Function codes (for those using synchronous interfaces) may be sent using the form '^=x', where 'x' is the function code you wish to be sent with the next message to be transmitted.

If the last character in the macro string typed in is '^', it will be ignored. This can in fact be useful: in that trailing spaces are deleted from the input you type; you can use an appropriately placed '^' at the end of the string to ensure that required trailing spaces are actually included. For example, entering 'ABC ^' will provide a trailing space after the 'ABC'.

Screen Options
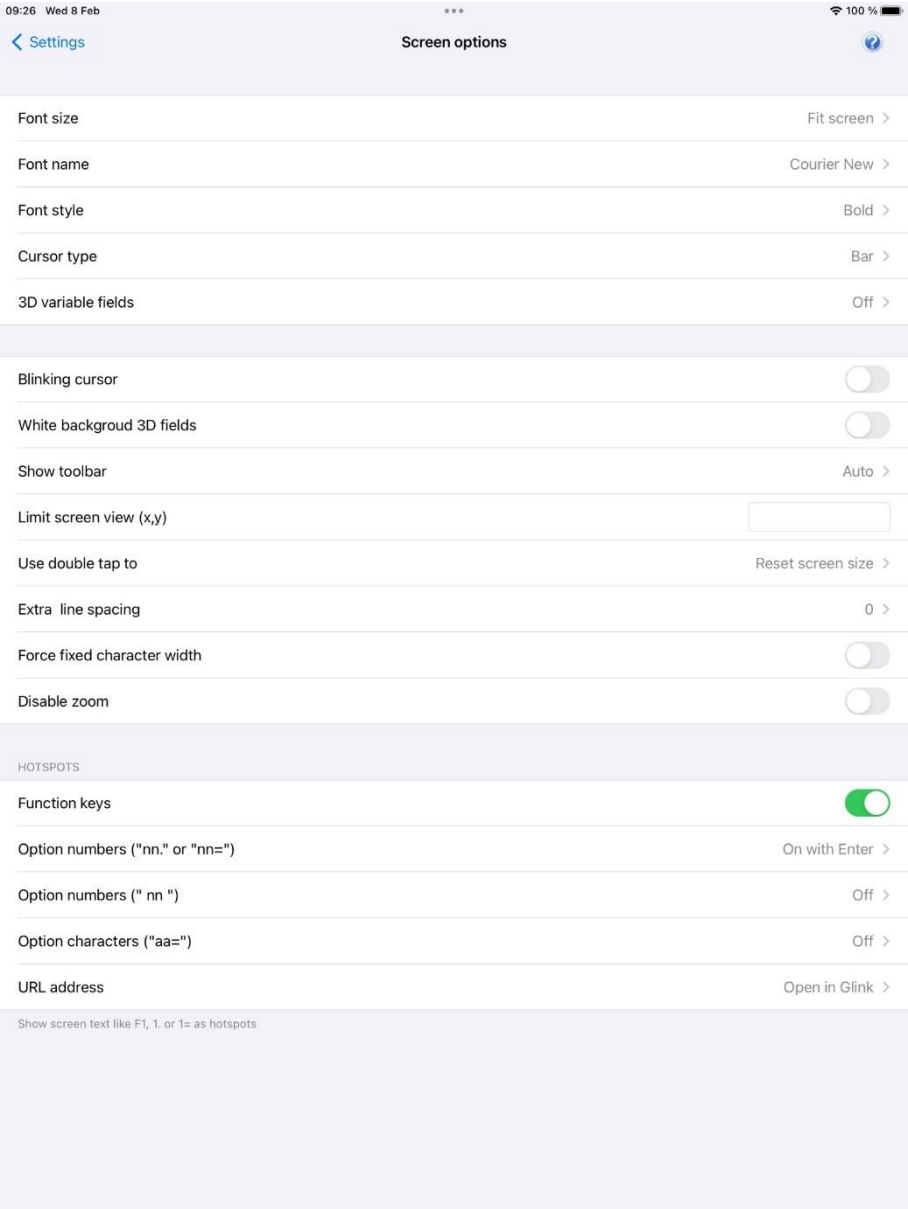
Font size

This option defines the initial font size to be used when Glink is started. If a large font is selected, some part of the screen display may not be visible without scrolling to it.

The *Auto* value selects a font size that is readable and, in most cases, suitable for the screen size.

The *Fit screen* options adjust the Glink display to fit the screen size. To do that, the characters will be stretched or squeezed.

Font name

This option defines the name of the font used in the emulator window. Only mono spaced fonts are available.

Font style

Here you may choose the display style for your chosen font *(iOS only)*.

Cursor type

The cursor indicates the current screen position for the emulator (as opposed to the mouse cursor, which operates independently). This option is used to choose one of the three available cursor types — a block cursor, a horizontal line cursor, or a vertically bar cursor.

3D variable fields

This option enables use of 3D effects for variable fields for those emulations that support the use of forms. You have four options, "On without attributes", "On with attributes", "On without underline" and "Show as underline". Normally a 3D field has white background, but you may keep the initial attributes set by the host for those input fields. The attributes can for example be specific background color or underlined text.

Blinking cursor

Here you may choose the cursor to blink. Please have in mind that a blinking cursor requires more resources, so screen updating, and scrolling might look better without it.

White background 3D fields

This option forces the background of a 3D variable field to always be white regardless of the default background color selected.

Similar, if the host specific sets the background color for an input field, the background color will still be white.

if you want the foreground color to always be black, select the 3D variable fields option "On without attributes".

Show toolbar

The toolbar can be shown as a horizontal single-line toolbar under the emulation screen with a toggle button to switch between the toolbars defined or as a horizontal multiline toolbar where all the toolbars defined are shown. If the multiline toolbar option is selected, Glink will still use the single-line toolbar to save space when the keyboard is up. For big screens operating in landscape mode, the toolbars can be shown vertically on the right side of the emulation screen if the vertical toolbar option is selected.

The *Auto* value means *Multiline* for a big screen (for example tablet) and *Single-line* for a small screen (for example phone).

The *Off* value means that the toolbar is not shown by default.

The *Vertical* value means that the toolbar is shown on the right side of the screen. The number of toolbars configured will decide the default number of toolbar columns, but the toolbar buttons will be distributed from left to right from the top and down, so that the *Toolbar 1* buttons will be shown on the first rows, then the *Toolbar 2* buttons will follow on the next rows and so on.

A Multiline toolbar like this:

←	↕	↕	→	Esc	Ctrl	→	Remove	Enter
PF1	PF2	PF3	PF4	F5	F6	F7	F8	F9	F10
F11	F12	F13	F14	Help	Do	F17	F18	F19	F20
Find	Select	PrevS	NextS	Del				

Will look like this when the vertical option is selected:

←	↕	↕	→
Esc	Ctrl	→	Remove
Enter		PF1	PF2
PF3	PF4	F5	F6
F7	F8	F9	F10
F11	F12	F13	F14
Help	Do	F17	F18
F19	F20	Find	Select
PrevS	NextS	Del	

For the value *Horizontal*, the toolbar is shown at the bottom of the screen using horizontal page scrolling.

Note that you can change the number of buttons and the number of toolbars in *Settings -> Toolbar* setup. Here you can Add toolbars, Add buttons, Modify buttons, Move buttons, Remove buttons, adjust the button size by the use of scaling factors and configure the Vertical or Horizontal toolbar layout.

Limit screen view (x,y)

If a host application uses just a part of the screen, the screen view may be decreased to match this size. The right and lower part will just be cut off and not visible. For example, if the parameter is set to "40,20", only the first 40 characters of each line up to line 20 will be visible. This parameter has no impact on the emulation itself. "x" limits only the number of columns, ",y" limits only the number of rows.

Use double tap to

This option specifies the action performed when double tapping the Glink screen, either *Reset screen size*, *Perform Enter* or *Toggle Menu bar On/Off*.

The *Reset screen size* sets the initial screen size, the *Perform Enter* does an Enter (Send) command and the *Toggle Menu bar On/Off (On)/Toggle Menu bar On/Off (Off)* toggles the menu bar *On* or *Off*. In the latter case, the menu bar is initially hidden. When the menu bar is hidden, it can be handy to add an Enter button to the toolbar (acts as a connect command if the session is disconnected)..

Extra line spacing

Add extra space between the lines (gives more space for hotspots if used). A negative number will decrease the standard line spacing.

Force fixed character width

Glink uses a fixed font, but for some special characters and symbols, the width may still vary. This option forces Glink to update the screen character by character at the correct position. Might slow down the screen update (Android option only).

Hotspots

The hotspot options tell Glink to search for specific keywords on the screen and highlight these as push buttons and assign a function to each of them.

For example, if the F6 text is on the screen, it will be highlighted as a push button and the function F6 will be assigned to it and executed when tapped.

If Option numbers is enabled, text like 2. or 2= will be searched for and highlighted (the number only). If this hotspot is tapped, the number is added at the current cursor position. Optionally it can be followed by the Enter function to transmit it. Numbers larger than 99 is not converted to hotspots.

If the next Option numbers is enabled, numbers that are both proceeded and followed by a space will be searched for and highlighted. If this hotspot is tapped, the number is added at the current cursor position and optionally followed by an Enter function to transmit it. Numbers larger than 99 is not converted to hotspots.

If Option characters is enabled, characters followed by the = character will be highlighted. If this hotspot is tapped, the character(s) is added at the current cursor position and optionally followed by an Enter function to transmit it. For example, S will be regarded as a hotspot for the text S=Search.

If URL address is enabled, URLs will be highlighted. If this hotspot is tapped, Glink will either open an external browser or use the built-in browser in Glink to display the URL content.

Screen Colors
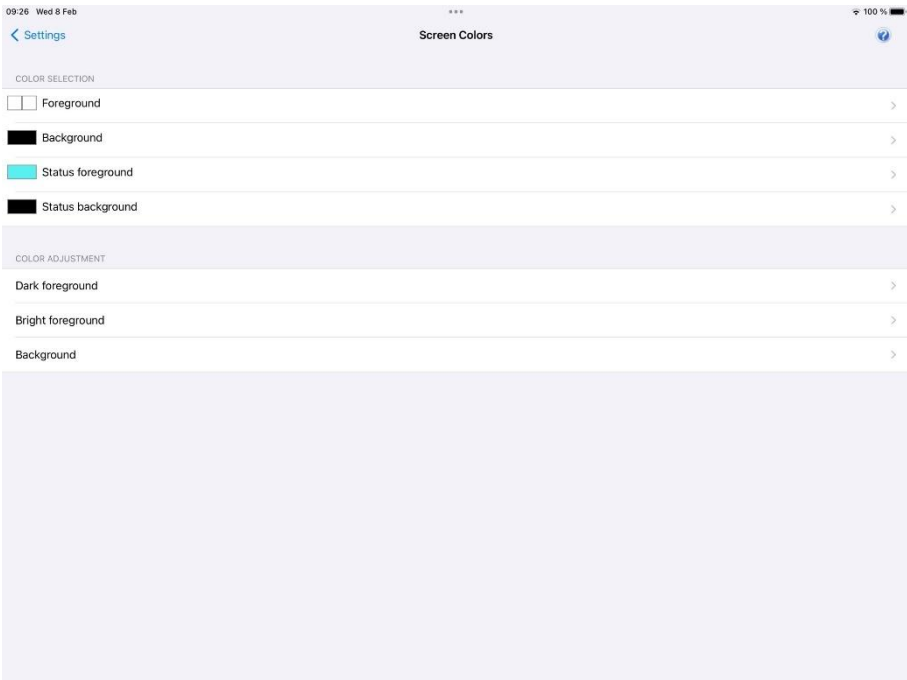
Color selection

The screen color options allow you to choose the default foreground and background colors for the emulator screen and for the status line.

The default name of the colors available are listed together with icon. If the color name does not match the color icon, the color has been adjusted.

Color adjustment

The color adjustment options allow you to adjust the 8 dark plus 8 bright foreground colors and the 8 background colors. The original color's names are displayed in the color adjustment display. If a color has been adjusted, it may not match the name for that color.

VIP attribute mapping
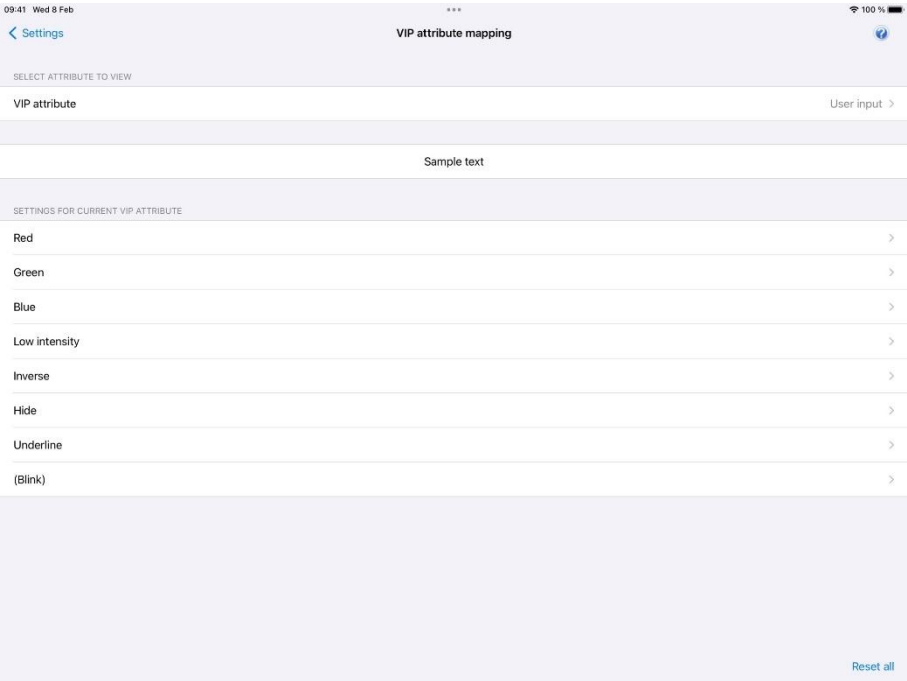
This menu allows you to select which visual attributes are to be associated with every type of field setting from the host. Each attribute defined by the host has a specific letter associated with it.

For each of the attributes that may be set by the host, you can specify which of the visual attributes should be associated with it. Each of the visual attributes may be left **as is** (leave blank), turned **ON**, or turned **OFF**. As an example: you wish all unprotected fields to be presented in red inverse video. You would first select the VIP attribute Unprotected at the top, then set Red ON, Green and Blue Off and last Inverse On. Note that colors are mixed, so to produce yellow for example, you would turn red and green ON, while turning blue OFF.

One additional VIP attribute is available and is listed as user input. This attribute will be applied to all user input typed at the keyboard so long as no attribute has been set at the current position by the host computer. This may be used to provide easy differentiation of text you yourself have typed and text that came from the host. (Note: this does not apply to ECHO mode - in ECHO mode all characters displayed come from the host). This specific attribute is not part of the standard VIP attribute set, but may be accessed by the host using the sequence <esc>s[.

Obviously, some of the attributes will already be set for you, for example the VIP attribute Inverse will be set by default to turn on inverse while leaving the other attributes alone. In the example above, if you wanted to ensure that the ONLY inverse video fields to be shown were the unprotected ones, then you should reset (clear) the settings for the VIP attribute Inverse.

ANSI / VT / DKU / 7700 / 7102 attribute mapping
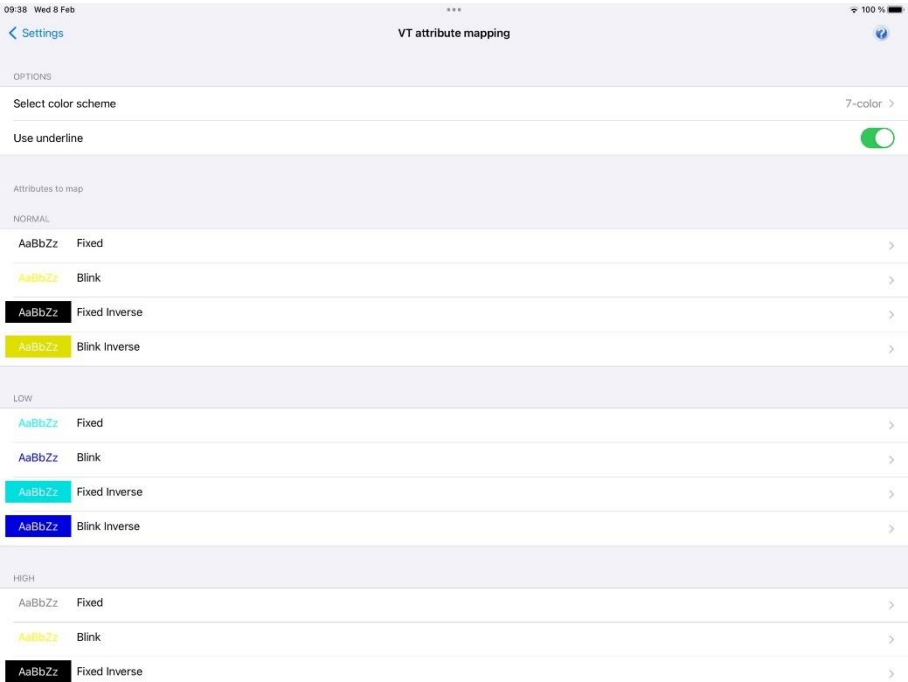
Settings – Screen

This configuration table allows you to select which attributes should be used in practice for various combinations of attributes sent by the host application.

There are four different color schemes available. In addition, a Custom color is available if you have modified one of the default ones.

Normally Glink will preserve the state of the underline as it is sent by the host, but if you have mapped underline to a specific color then you may find that to be enough. In that case uncheck the Use underline option.

You may modify the color scheme selected to create your own one. Click on the attribute field to remap. Then choose the desired foreground and background colors from the list boxes. Repeat these operations for the other attribute fields to be remapped.

If you have modified one of the existing colors schemes, it will be saved as Custom colors scheme. If custom colors scheme already exists, it will be overwritten.

Settings – Devices

Print options
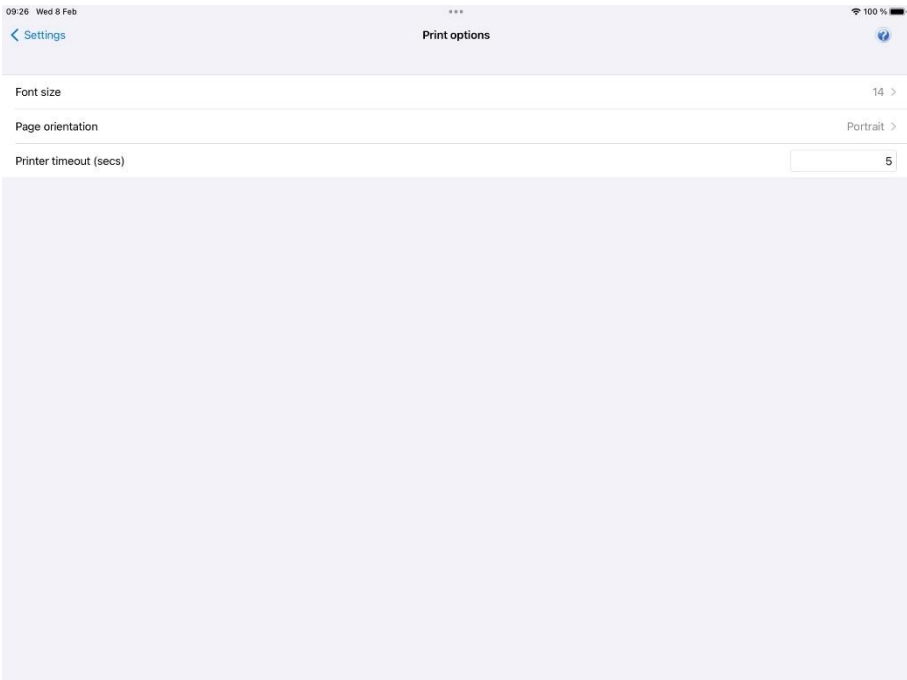
Here you can adjust the Font size and Page orientation so that the print data will have the desired look when printed.

Tap the Print data button to return to the print data screen.

Barcode reader setup

Barcode reader device type

Barcode reader device type is either *Keyboard Wedge* or *Keyboard Wedge A*. The barcode data is inserted as normal input data at the current cursor location of the Glink display.

When the barcode scanner is used as a HID keyboard (Keyboard Wedge) input device, Glink will not know if the data is coming from the actual keyboard or from the scanner. The "More options" described below is not in use for scanners operating in *Keyboard wedge* mode.

If you want to use these options in Glink, select *Keyboard Wedge* A(dvanced) mode and configure the barcode scanner to send the prefix and suffix described below in the Keyboard Wedge A section.

`Key-`
`board`
`Wedge`
	Default. Scanned data is received as keyboard in-put. Glink cannot separate scanned data from keyboard input.

`Key-`
`board`
`Wedge A`
	Scanned data is still received as keyboard input, but if special prefix and suffix characters is con-figured, use Keyboard Wedge A(dvanced) to ena-ble use the options below.

Keyboard Wedge A

The Keyboard Wedge A(dvanced) mode requires that the barcode scanner (not Glink) is set up with hex *0E (SO)* as prefix character and hex *0F (SI)* as suffix char-acter. Some devices let you select *SO (shift out)* as the prefix and *SI (shift in)* as the suffix, others let you configure the hex value. For others it is not possible to specify these low ascii characters, or low ascii characters are stripped off. In this case, spec-ify the prefix as the string *^$0E* and the suffix as the string *^$0F*.

If the prefix and the suffix are not set up properly, the default Keyboard Wedge mode will be used instead.

When the Keyboard Wedge A(dvanced) mode is selected, the options described un-der More options below are available.

To check that Keyboard Wedge A(dvanced) is configured correctly, enable the op-tion *Show on status line* to have the scanned data displayed in inverse text on the sta-tus line at the bottom of the Glink screen.

More options

Please note that most scanner software can be configured with options like the ones below. Make sure that these options are either set in Glink or in the Barcode Scanner settings itself. Check the Barcode Scanner manual.

Action after scan

```
None      Do no action after scanning
Tab       Do Tab after scanning
Enter     Do Enter after scanning
FldExit   Do Field Exit after scanning (5250 emulation mode only)
```

Remove char(s) at start

Removes one or more characters at the beginning of the barcode.

Remove char(s) at end

Removes one or more characters at the end of the barcode.

Add text before

Add text before the barcode.

Add text after

Add text after the barcode (the Action after scan will be added after this text).

Use keyboard mapping

Enable this option if keyboard mapping should be applied for scanned data. For example, if colon (:) is mapped to the *Tab* function and the semicolon (;) is mapped to the *Enter* function, a scanned barcode "1234:5678;" will be interpreted as key input 1234**Tab**5678**Enter.** Another example can be that you want to remove (Ignore) the Group Separator (GS - hex 1D) character in a barcode

Show on status line

Enable this option to display the scanned data as a status message at the bottom of the screen as well.

Keyboard options

Glink uses any of the standard iOS keyboards configured. The keyboard configurations are found in the iOS Settings app in the **General-> Keyboard** section.

External keyboard

When using an external keyboard, in the Glink main screen, select *Menu-> General options-> Keyboard up on*. Glink will then be ready for keyboard input when connected to a host application (otherwise you must tap at an input field first).

Glink support several keyboard shortcuts for function and control keys. To display a list of these shortcuts, press and hold the *cmd* key until the list is displayed.
See Keyboard Shortcuts in the *Getting started* section.

Keys can be mapped or remapped, select Setting->Keyboard mapping option.

Keyboard mapping
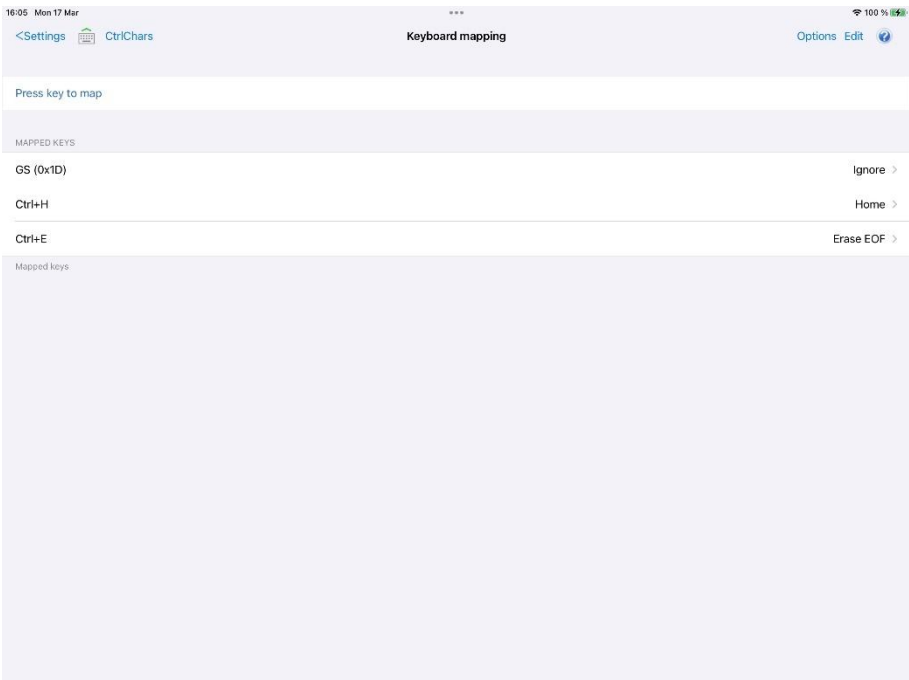
Settings – Devices

Glink is initially set up with a default keyboard mapping when using an external keyboard. You may override the default mapping or add a new mapping.

The list box displays the mapped keys if any.

Tap **CtrlChars** to map a low ASCII control character. You will get a list of control characters from hex 00 (0x00) to hex 1F (0x1F). This is useful if you for example want to remove the GS (0x1D) (Group Separator) character from a scanned barcode.

Press the key to map. Select the keyboard icon button to bring up the soft keyboard if needed. Once a key is pressed and confirmed to map, a new row is added. Select the *Key not mapped* entry to map the key to a function or macro. If a macro is selected, select the **Detail disclosure button** to define the actual macro.

Select the **Edit** option to remove a key mapping.
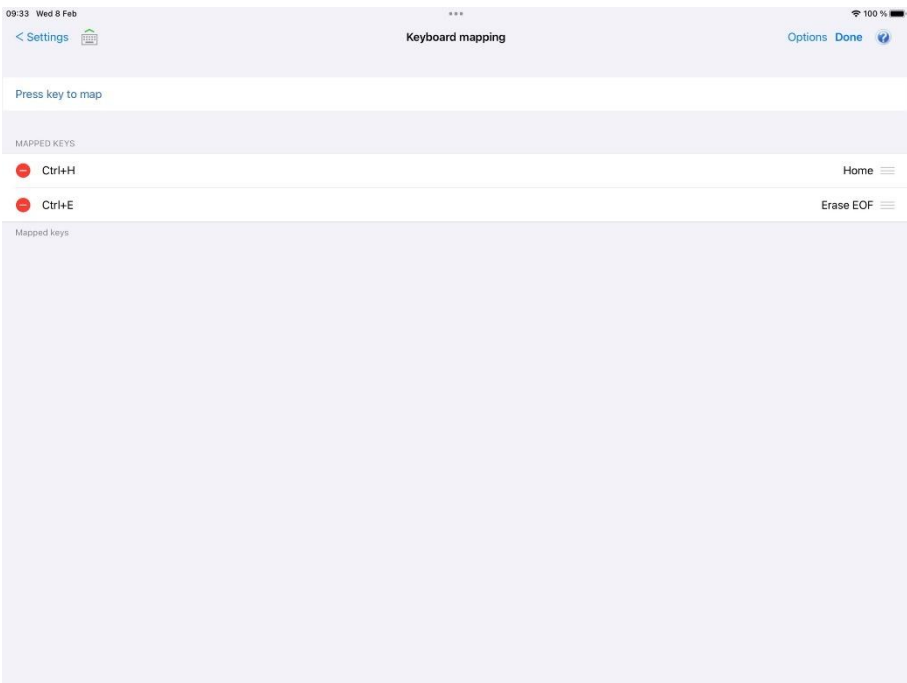
Please note that prior iOS 13.4 various keys on the external keyboard (like the function keys F1, F2...) are not supported by iOS and cannot be mapped.

Define macro

The macro must be defined and the *Macro string* field below defines the macro.

Button text
The button text is information only and is not used when the macro is executed.

Macro string
When entering a macro string, in addition to plain text, there are several conventions, all of which are signaled using the caret (^). If you wish to enter a 'real' caret then you must type it twice (^^).

Control characters may be entered using the normal convention with a letter following the caret symbol. For example, a return is ^M and a line feed is ^J.

You may also send specific ASCII codes in hexadecimal, decimal, or octal form by following the caret with a $, #, or & character, and the desired code:
`^#ddd` decimal specification
`^&ooo` octal specification
`^$hh`   hexadecimal specification

Note that decimal and octal codes must be 3 digits and prefixed with 0 if it is 2 digit codes.

Macro string tip
Use the Record toolbar macro function to define the required macro and then copy the macro string generated and insert it as *Macro string*. The generated string is found in Setting->Toolbar setup. Select the macro and then the Modify function found in the drop-down Action list.

Distributing Glink configurations from an EMM/MDM

Glink has many configuration parameters that are available through the *Settings* screens in Glink.

A Glink configuration can be prepared and tested on one device, then exported from this device and imported by Glink on other devices. If your MDM does not allow you to distribute the exported Glink configuration file as it is, it will probably allow you to distribute Glink parameters to a managed Enterprise version of Glink, either using a menu to set the Glink parameters or by importing a JSON formatted configuration file containing the Glink parameters. Either way, you will find the parameters in the exported Glink configuration file very useful when setting up such a configuration. Make a note of which parameters you have set/changed, because these are the parameters you must configure and distribute from your MDM. It is not necessary to distribute default parameter values.

Below you will find some examples on how you can distribute a Glink configuration file.

Download the Glink configuration from a web server

The managed versions of Glink (all iOS versions and Glink VT/5250/3270 Enterprise for Android) can be configured directly from the MDM if the MDM supports managed app configuration. The MDM either use a menu to set the Glink parameters or it imports a JSON formatted configuration file containing the Glink parameters. The method and format vary with the different MDMs.

The MDM server can instruct Glink to import a Glink configuration file from a network location with the following parameters:

config.version	This parameter (key-name) specifies the version of the MDM parameters. If the version number is equal to the version number on the device, the rest of the parameters are ignored by Glink.

con-fig.url	This parameter (key-name) provides the URL of the Glink configuration file. Please note that this parameter is ignored if the *config.version* parameter is missing or the version number is the same as the version number on the device. So every time you update the configuration file, you must remember to update the *config.version* parameter

Here is an example:

```
config.version=2
config.url=http://myhost.mydomain.xx/glink/group1/config.glinki
```

Depending on the MDM, these two parameters are either configured in a menu in the MDM console (MobileIron, Meraki, Intune) or in a JSON file uploaded to the MDM (Google Admin) or defined in an MDM-menu (Intune).

Here is an example from Microsoft Intune (Android):
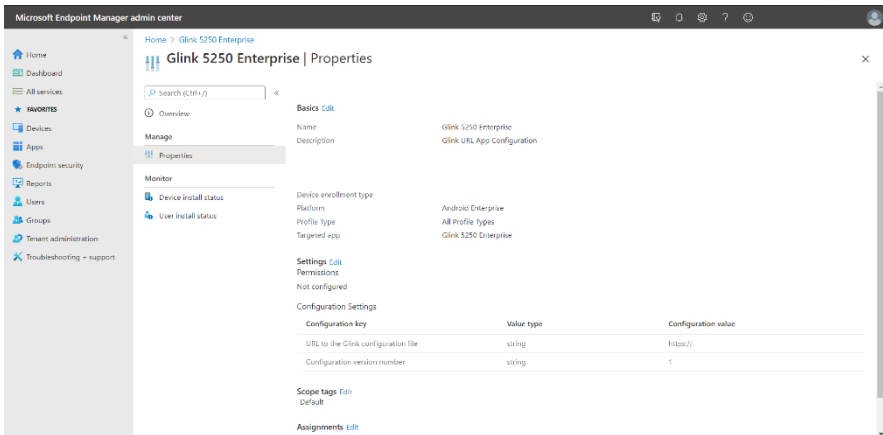
Set Glink parameters from the MDM

The iOS versions and the managed Glink Enterprise for Android versions of Glink (Glink 5250 Enterprise, Glink VT Enterprise, Glink 3270 Enterprise and Glink Enterprise) can be configured directly from the MDM if the MDM supports managed app configuration. The MDM either use a menu to set the Glink parameters or it imports a JSON formatted configuration file containing the Glink parameters. The method and format vary with the different MDMs.

Distributing Glink configurations from an EMM/MDM

Here are three key parameters that are explicitly defined with information texts and that are the minimum set of parameters required for a host session configuration:

config.ver-sion	This parameter (key-name) specifies the version of the MDM parameters. If the version number is equal to the version number on the device, the rest of the parameters are ignored.
con-fig.name	This is the name of the configuration (session) and must be provided. The configuration is created if it does not already exist. Additional parameters will be set for this selected configuration.
com.server-name	This specifies the IP address of your host

Here is an example:

```
config.version=1
config.name=myhost
com.servername=myhost.mydomain.com
```

These parameters are sufficient to create a host configuration with default values. For other parameters you have to check your exported Glink configuration file, *config.glinki*. These parameters must be defined as parameter_key /parameter_value pairs.

For example:

Parameter key *1_config.name* / **Parameter value** *myhost2*
Parameter key *1_com.servername* / **Parameter value** *host2.mydomain.com*
Parameter key *2_config.name* / **Parameter value** *myhost3*
Parameter key *2_com.servername* / **Parameter value** *host3.mydomain.com*

Parameters for the first session, named with *config.name*, you do not need a prefix, but other sessions need a prefix. Session 2 parameters are prefixed with **1_**, session 3 parameters are prefixed with **2_** and so on.

How you set these parameters will depend on your MDM. Different methods are used. While some MDMs present a menu with the parameters that can be set (MobileIron, Meraki, Intune), others require a JSON-/XML-file with key-name and key-value pairs (Google Admin, Intune).

Here are some examples:

MobileIron (Android):

MobileIron (iOS):

Distributing Glink configurations from an EMM/MDM

Meraki (Android):

Meraki (iOS):

General options

Parameters that are common to all sessions and configured in the *General options* menu, must be configured in a *config.name* named *GeneralOptions*, **without** a *n_* prefix.

For example:

```
Parameter key config.name / Parameter value GeneralOptions
Parameter key screen.noautolock / Parameter value true
```

Passwords

Passwords are normally entered as the encrypted string you see in an exported Glink configuration file. Starting from version 2.4.3 you can also specify passwords in clear text if you use the prefix *##*. For example, the password that protects access to *Settings*, *General options*, *New configuration*, *Export configurations* and more:

```
Parameter key config.name / Parameter value GeneralOptions
Parameter key screen.cfgpassword / Parameter value ##settingpassword
```

Distributing Glink configurations from an EMM/MDM

```
or the password for an auto-logon to IBM AS400/iSeries:
```

Parameter key *config.name* / **Parameter value** *AS400*
Parameter key *com.servername* / **Parameter value** *myhost.mydomain.com*
Parameter key *com.ibm5250user* / **Parameter value** *username*
Parameter key *com.ibm5250subspw* / **Parameter value** *##as400password*
Parameter key *com.ibm5250usealtlogin* / **Parameter value** *true*

Toolbar

If you want to configure your own toolbar(s), you might want to remove the default toolbars with all their buttons before you start defining your own. To do that, you can use the parameter *toolbar.remove* and set it to *true*. Then you can add your own buttons, for example like this:

Parameter key *toolbar.remove* / **Parameter value** *true*
Parameter key *toolbar.cmd0* / **Parameter value** 54,Left,,\u21a4
Parameter key *toolbar.cmd1* / **Parameter value** 51,Up,,\u21a5
Parameter key *toolbar.cmd2* / **Parameter value** 52,Down,,\u21a7
Parameter key *toolbar.cmd3* / **Parameter value** 53,Right,,\u21a6

Remove a session configuration

If you want to remove a session (*config.name*) from your configuration, you can use the parameter *config.remove*, for example:

Parameter key *config.remove* / **Parameter value** *AS400*

XML Schema for the iOS version

The AppConfig Community, is sponsored by industry leading device management vendors, have defined a standard schema to configure apps. Using this definition, the Glink parameters can be specified in an XML document using the following syntax:

```
<managedAppConfiguration>
<version>1</version>
<bundleId>no.gar.glink5250</bundleId>
<dict>
    <string keyName="config.name">
        <defaultValue>
                <value>MyConfigName</value>
        </defaultValue>
    </string>
    <string keyName="com.servername">
        <defaultValue>
                <value>MyServerAddress</value>
        </defaultValue>
    </string>
</dict>
</managedAppConfiguration>
```

Please note that the **bundleId** must match the Glink version you are using; either no.gar.glinkvt, no.gar.glink5250, no.gar.glink3270, no.gar.glink for the standard Glink versions or no.gar.eglinkvt, no.gar.eglink5250, no.gar.eglink3270 or no.gar.eglink for the Enterprise versions of Glink.

Example XML Schema Glink VT for iOS

Below you will find an example XML schema for Glink VT Enterprise for iOS for a SSH connection to *myserver1.domain.xx*.

Note that to set the General options parameters the *config.name* parameter must have the value *GeneralOptions*. Parameters for the first *Session* are prefixed with *1_*.

The example also includes the use of the *config.url* parameter with value: *http://mywebserver.domain.xx/glinkvt/group1/config.glinki*, which you can use to instruct Glink VT to download a Glink configuration file from a webserver.

```
<managedAppConfiguration>
<version>1</version>
<bundleId>no.gar.eglinkvt</bundleId>
<dict>
  <integer keyName="config.version">
```

```
    <defaultValue>
      <value>1</value>
    </defaultValue>
  </integer>
  <string keyName="config.url">
    <defaultValue>
      <value>http://mywebserver.domain.xx/glinkvt/group1/con-
fig.glinki</value>
    </defaultValue>
  </string>
  <string keyName="config.name">
    <defaultValue>
      <value>GeneralOptions</value>
    </defaultValue>
  </string>
  <boolean keyName="screen.noAutoLock">
    <defaultValue>
      <value>false</value>
    </defaultValue>
  </boolean>
  <boolean keyName="screen.autoConnect">
    <defaultValue>
      <value>false</value>
    </defaultValue>
  </boolean>
  <string keyName="screen.showKeyboard">
    <defaultValue>
      <value>0</value>
    </defaultValue>
    <constraint>
      <values>
        <value>0</value>
        <value>1</value>
        <value>2</value>
      </values>
    </constraint>
  </string>
   <string keyName="1_config.name">
    <defaultValue>
      <value>myserver1</value>
    </defaultValue>
  </string>
  <boolean keyName="1_emu.allowLcase">
    <defaultValue>
      <value>false</value>
    </defaultValue>
  </boolean>
  <string keyName="1_emu.type">
    <defaultValue>
      <value>VT</value>
    </defaultValue>
  </string>
  <boolean keyName="1_emu.autoTabL">
    <defaultValue>
      <value>false</value>
    </defaultValue>
```

```
    </boolean>
    <string keyName="1_screen.foregroundMagentaLo">
      <defaultValue>
        <value>#ff00ff</value>
      </defaultValue>
    </string>
    <string keyName="1_screen.foregroundRedLo">
      <defaultValue>
        <value>#f01818</value>
      </defaultValue>
    </string>
    <string keyName="1_screen.foregroundBlueLo">
      <defaultValue>
        <value>#7890f0</value>
      </defaultValue>
    </string>
    <string keyName="1_screen.fontName">
      <defaultValue>
        <value>Courier</value>
      </defaultValue>
    </string>
    <integer keyName="1_screen.variable3Ds">
      <defaultValue>
        <value>1</value>
      </defaultValue>
    </integer>
    <integer keyName="1_screen.scrColor">
      <defaultValue>
        <value>3</value>
      </defaultValue>
    </integer>
    <integer keyName="1_screen.stsColor">
      <defaultValue>
        <value>3</value>
      </defaultValue>
    </integer>
    <string keyName="1_screen.foregroundYellowLo">
      <defaultValue>
        <value>#ffff00</value>
      </defaultValue>
    </string>
    <string keyName="1_screen.foregroundWhiteLo">
      <defaultValue>
        <value>#ffffff</value>
      </defaultValue>
    </string>
    <string keyName="1_screen.foregroundCyanLo">
      <defaultValue>
        <value>#58f0f0</value>
      </defaultValue>
    </string>
    <string keyName="1_screen.foregroundGreenLo">
      <defaultValue>
        <value>#24d830</value>
      </defaultValue>
```

```
    </string>
    <string keyName="1_com.type">
      <defaultValue>
        <value>SSH</value>
      </defaultValue>
    </string>
    <string keyName="1_com.serverName">
      <defaultValue>
        <value>myserver1.domain.xx</value>
      </defaultValue>
    </string>
    <boolean keyName="1_com.usessh">
      <defaultValue>
        <value>true</value>
      </defaultValue>
    </boolean>
    <string keyName="1_com.sshUser">
      <defaultValue>
        <value>Username</value>
      </defaultValue>
    </string>
    <string keyName="1_com.sshPassword">
      <defaultValue>
        <value>##Password</value>
      </defaultValue>
    </string>
  </dict>
  <presentation defaultLocale="en-US">
      <field keyName="config.version" type="input">
          <label>
              <language value="en-US">Configuration version num-
ber</language>
          </label>
          <description>
              <language value="en-US">The version number must be
present and different from the current version for the Configuration
URL to take</language>
          </description>
      </field>
      <field keyName="config.url" type="input">
          <label>
              <language value="en-US">URL of Configuration
file</language>
          </label>
          <description>
              <language value="en-US">This parameter provides the
URL of the Glink configuration file. Please note that this parameter
is ignored if the config.version parameter is missing or the version
matches the version on the device</language>
          </description>
      </field>
    <fieldGroup>
      <name>
        <language value="en-US">GeneralOptions</language>
      </name>
        <field keyName="config.name" type="input">
```

```
            <label>
                <language value="en-US">Configuration name GeneralOp-
tions</language>
            </label>
            <description>
                <language value="en-US">Configuration name GeneralOp-
tions is required for setting parameters in General Options</language>
            </description>
        </field>
        <field keyName="screen.noAutoLock" type="checkbox">
            <label>
                <language value="en-US">Never Auto-Lock when con-
nected</language>
            </label>
            <description>
                <language value="en-US">Disable iOS Auto-Lock feature
when connected</language>
            </description>
        </field>
        <field keyName="screen.autoConnect" type="checkbox">
            <label>
                <language value="en-US">Auto-connect at start-up</lan-
guage>
            </label>
            <description>
                <language value="en-US">Auto-connect first session at
start-up</language>
            </description>
        </field>
  <field keyName="screen.showKeyboard" type="select">
    <label>
      <language value="en-US">Keyboard up</language>
    </label>
    <description>
      <language value="en-US">By default Glink will show the keyboard
when you click an input field and hide it if you click outside. If you
select On, an external keyboard will always be connected if you have
one or the keyboard will always be shown</language>
    </description>
    <options>
      <option selected="true" value="0">
        <language value="en-US">Default</language>
      </option>
      <option value="1">
        <language value="en-US">On</language>
      </option>
      <option value="2">
        <language value="en-US">Off</language>
      </option>
    </options>
  </field>
        <field keyName="1_config.name" type="input">
            <label>
                <language value="en-US">Configuration name for ses-
sion</language>
```

```
            </label>
            <description>
                <language value="en-US">Name of session configura-
tion</language>
            </description>
        </field>
    </fieldGroup>
  <fieldGroup>
      <name>
        <language value="en-US">Emulation options</language>
      </name>
      <field keyName="1_emu.allowLcase" type="input">
        <label>
          <language value="en-US">emu.allowLcase</language>
        </label>
        <description>
          <language value="en-US">Allow lowercase to host</language>
        </description>
      </field>
      <field keyName="1_emu.type" type="input">
        <label>
          <language value="en-US">emu.type</language>
        </label>
        <description>
          <language value="en-US">Emulation Mode</language>
        </description>
      </field>
      <field keyName="1_emu.autoTabL" type="input">
        <label>
          <language value="en-US">emu.autoTabL</language>
        </label>
        <description>
          <language value="en-US">Auto tab when input field is
filled</language>
        </description>
      </field>
    </fieldGroup>
    <fieldGroup>
      <name>
        <language value="en-US">Screen options</language>
      </name>
      <field keyName="1_screen.foregroundMagentaLo" type="input">
        <label>
          <language value="en-
US">screen.foregroundMagentaLo</language>
        </label>
        <description>
          <language value="en-US">Magenta</language>
        </description>
      </field>
      <field keyName="1_screen.foregroundRedLo" type="input">
        <label>
          <language value="en-US">screen.foregroundRedLo</language>
        </label>
        <description>
          <language value="en-US">Red</language>
```

```
      </description>
    </field>
    <field keyName="1_screen.foregroundBlueLo" type="input">
      <label>
        <language value="en-US">screen.foregroundBlueLo</language>
      </label>
      <description>
        <language value="en-US">Blue</language>
      </description>
    </field>
    <field keyName="1_screen.fontName" type="input">
      <label>
        <language value="en-US">screen.fontName</language>
      </label>
      <description>
        <language value="en-US">Font name</language>
      </description>
    </field>
    <field keyName="1_screen.variable3Ds" type="input">
      <label>
        <language value="en-US">screen.variable3Ds</language>
      </label>
      <description>
        <language value="en-US">3D variable fields</language>
      </description>
    </field>
    <field keyName="1_screen.scrColor" type="input">
      <label>
        <language value="en-US">screen.scrColor</language>
      </label>
      <description>
        <language value="en-US">Screen color</language>
      </description>
    </field>
    <field keyName="1_screen.stsColor" type="input">
      <label>
        <language value="en-US">screen.stsColor</language>
      </label>
      <description>
        <language value="en-US">Status line color</language>
      </description>
    </field>
    <field keyName="1_screen.foregroundYellowLo" type="input">
      <label>
        <language value="en-US">screen.foregroundYellowLo</language>
      </label>
      <description>
        <language value="en-US">Yellow</language>
      </description>
    </field>
    <field keyName="1_screen.foregroundWhiteLo" type="input">
      <label>
        <language value="en-US">screen.foregroundWhiteLo</language>
      </label>
      <description>
```

```
        <language value="en-US">White</language>
      </description>
    </field>
    <field keyName="1_screen.foregroundCyanLo" type="input">
      <label>
        <language value="en-US">screen.foregroundCyanLo</language>
      </label>
      <description>
        <language value="en-US">Cyan</language>
      </description>
    </field>
    <field keyName="1_screen.foregroundGreenLo" type="input">
      <label>
        <language value="en-US">screen.foregroundGreenLo</language>
      </label>
      <description>
        <language value="en-US">Green</language>
      </description>
    </field>
  </fieldGroup>
  <fieldGroup>
    <name>
      <language value="en-US">Communication options</language>
    </name>
    <field keyName="1_com.type" type="input">
      <label>
        <language value="en-US">com.type</language>
      </label>
      <description>
        <language value="en-US">Comms interface</language>
      </description>
    </field>
    <field keyName="1_com.serverName" type="input">
      <label>
        <language value="en-US">com.serverName</language>
      </label>
      <description>
        <language value="en-US">Server address</language>
      </description>
    </field>
    <field keyName="1_com.usessh" type="input">
      <label>
        <language value="en-US">com.usessh</language>
      </label>
      <description>
        <language value="en-US">Use SSH connection</language>
      </description>
    </field>
    <field keyName="1_com.sshUser" type="input">
      <label>
        <language value="en-US">com.sshUser</language>
      </label>
      <description>
        <language value="en-US">User name:</language>
      </description>
    </field>
```

```
      <field keyName="1_com.sshPassword" type="input">
        <label>
          <language value="en-US">com.sshPassword</language>
        </label>
        <description>
          <language value="en-US">Password:</language>
        </description>
      </field>
    </fieldGroup>
  </presentation>
</managedAppConfiguration>
```

Example .plist file for Glink VT for iOS

Below you will find an example .plist-file for Glink VT Enterprise for iOS for a
SSH connection to *myserver1.domain.xx*.

Note that to set the General options parameters the *config.name* parameter must have
the value *GeneralOptions*. Parameters for the first *Session* are prefixed with *1_*.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.ap-
ple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>config.version</key>
    <integer>1</integer>
    <key>config.name</key>
    <string>GeneralOptions</string>
    <key>screen.noAutoLock</key>
    <true/>
    <key>1_config.name</key>
    <string>myserver1</string>
    <key>1_com.servername</key>
    <string>myserver1.domain.xx</string>
    <key>1_com.sshUser</key>
    <string>Username</string>
    <key>1_com.sshPassword</key>
    <string>##PASSWORD</string>
    <key>1_com.useSSH</key>
    <true/>
    <key>1_com.type</key>
    <string>SSH</string>
    <key>1_emu.type</key>
    <string>VT</string>
    <key>1_emu.autotabl</key>
    <false/>
    <key>1_screen.foregroundmagentalo</key>
    <string>#ff00ff</string>
    <key>1_screen.foregroundredlo</key>
    <string>#f01818</string>
    <key>1_screen.foregroundbluelo</key>
    <string>#7890f0</string>
    <key>1_screen.foregroundyellowlo</key>
    <string>#ffff00</string>
    <key>1_screen.foregroundwhitelo</key>
    <string>#ffffff</string>
    <key>1_screen.foregroundcyanlo</key>
    <string>#58f0f0</string>
    <key>1_screen.foregroundgreenlo</key>
    <string>#24d830</string>
    <key>1_screen.fontname</key>
    <string>Courier</string>
    <key>1_screen.variable3Ds</key>
    <integer>1</integer>
```

```
    <key>1_screen.scrColor</key>
    <integer>3</integer>
    <key>1_screen.stsColor</key>
    <integer>3</integer>
</dict>
</plist>
```

Example XML schema Glink 5250 for iOS

Below you will find an example XML schema for Glink 5250 Enterprise for iOS for a TN5250 connection to *myserver1.domain.xx*.

Note that to set the General options parameters the *config.name* parameter must have the value *GeneralOptions*. Parameters for the first *Session* are prefixed with *1_*.

The example also includes the use of the *config.url* parameter with value: *http://mywebserver.domain.xx/glink5250/group1/config.glinki*, which you can use to instruct Glink VT to download a Glink configuration file from a webserver.

```
<managedAppConfiguration>
    <version>1</version>
    <bundleId>no.gar.eglink5250</bundleId>
    <dict>
    <string keyName="config.version">
      <defaultValue>
        <value>1</value>
      </defaultValue>
    </string>
    <string keyName="config.url">
      <defaultValue>
        <value>http://mywebserver.domain.xx/glink5250/group1/con-
fig.glink5250</value>
      </defaultValue>
    </string>
    <string keyName="config.name">
      <defaultValue>
        <value>GeneralOptions</value>
      </defaultValue>
    </string>
    <boolean keyName="screen.noAutoLock">
      <defaultValue>
        <value>false</value>
      </defaultValue>
    </boolean>
    <boolean keyName="screen.autoConnect">
      <defaultValue>
        <value>false</value>
      </defaultValue>
    </boolean>
    <string keyName="screen.showKeyboard">
      <defaultValue>
        <value>0</value>
      </defaultValue>
      <constraint>
        <values>
          <value>0</value>
```

```
        <value>1</value>
        <value>2</value>
      </values>
    </constraint>
</string>
<string keyName="1_config.name">
  <defaultValue>
    <value>myhost1</value>
  </defaultValue>
</string>
<string keyName="1_com.servername">
  <defaultValue>
    <value>myserver1.domain.xx</value>
  </defaultValue>
</string>
<string keyName="1_com.ibm5250User">
  <defaultValue>
    <value>Username</value>
  </defaultValue>
</string>
<string keyName="1_com.ibm5250SUBSPW">
  <defaultValue>
    <value>##Password</value>
  </defaultValue>
</string>
<boolean keyName="1_com.ibm5250UseAltLogin">
  <defaultValue>
    <value>false</value>
  </defaultValue>
</boolean>
<string keyName="1_com.tn5250maindevice">
  <defaultValue>
    <value>#XX*======</value>
  </defaultValue>
</string>
<string keyName="1_com.ibm5250Model">
  <defaultValue>
    <value>0</value>
  </defaultValue>
  <constraint>
    <values>
      <value>0</value>
      <value>1</value>
      <value>2</value>
      <value>3</value>
      <value>4</value>
      <value>5</value>
      <value>6</value>
      <value>7</value>
      <value>8</value>
      <value>9</value>
      <value>10</value>
      <value>11</value>
    </values>
  </constraint>
```

```
    </string>
    <boolean keyName="1_com.useSSL">
      <defaultValue>
        <value>false</value>
      </defaultValue>
    </boolean>
    <boolean keyName="1_com.valServerCert">
      <defaultValue>
        <value>false</value>
      </defaultValue>
    </boolean>
    <string keyName="1_emu.ibmCodePage">
      <defaultValue>
        <value>500</value>
      </defaultValue>
      <constraint>
        <values>
          <value>037</value>
          <value>1140</value>
          <value>273</value>
          <value>1141</value>
          <value>1388</value>
          <value>880</value>
          <value>1025</value>
          <value>277</value>
          <value>1142</value>
          <value>278</value>
          <value>1143</value>
          <value>875</value>
          <value>280</value>
          <value>1144</value>
          <value>930</value>
          <value>930E</value>
          <value>939</value>
          <value>1399</value>
          <value>933</value>
          <value>1364</value>
          <value>870</value>
          <value>284</value>
          <value>1145</value>
          <value>285</value>
          <value>1146</value>
          <value>297</value>
          <value>1147</value>
          <value>500</value>
          <value>1148</value>
          <value>1149</value>
          <value>1026</value>
          <value>1155</value>
          <value>937</value>
          <value>1371</value>
        </values>
      </constraint>
    </string>
    <boolean keyName="1_emu.anyCmdResets">
      <defaultValue>
```

```
        <value>true</value>
      </defaultValue>
    </boolean>
    <boolean keyName="1_emu.errToStatBar">
      <defaultValue>
        <value>true</value>
      </defaultValue>
    </boolean>
    <boolean keyName="1_emu.anywhereOk">
      <defaultValue>
        <value>false</value>
      </defaultValue>
    </boolean>
  </dict>
<presentation defaultLocale="en-US">
        <field keyName="config.version" type="input">
            <label>
                <language value="en-US">Configuration version num-
ber</language>
            </label>
            <description>
                <language value="en-US">The version number must be
present and different from the current version for the Configuration
URL to take</language>
            </description>
        </field>
        <field keyName="config.url" type="input">
            <label>
                <language value="en-US">URL of Configuration
file</language>
            </label>
            <description>
                <language value="en-US">This parameter provides the
URL of the Glink configuration file. Please note that this parameter
is ignored if the config.version parameter is missing or the version
matches the version on the device</language>
            </description>
        </field>
        <field keyName="config.name" type="input">
            <label>
                <language value="en-US">Configuration name GeneralOp-
tions</language>
            </label>
            <description>
                <language value="en-US">Configuration name GeneralOp-
tions is required for setting parameters in General Options</language>
            </description>
        </field>
        <field keyName="screen.noAutoLock" type="checkbox">
            <label>
                <language value="en-US">Never Auto-Lock when con-
nected</language>
            </label>
            <description>
```

```
                    <language value="en-US">Disable iOS Auto-Lock feature
when connected</language>
            </description>
        </field>
        <field keyName="screen.autoConnect" type="checkbox">
            <label>
                <language value="en-US">Auto-connect at start-up</lan-
guage>
            </label>
            <description>
                <language value="en-US">Auto-connect first session at
start-up</language>
            </description>
        </field>
  <field keyName="screen.showKeyboard" type="select">
    <label>
      <language value="en-US">Keyboard up</language>
    </label>
    <description>
      <language value="en-US">By default Glink will show the keyboard
when you click an input field and hide it if you click outside. If you
select On, an external keyboard will always be connected if you have
one or the keyboard will always be shown</language>
    </description>
    <options>
      <option selected="true" value="0">
        <language value="en-US">Default</language>
      </option>
      <option value="1">
        <language value="en-US">On</language>
      </option>
      <option value="2">
        <language value="en-US">Off</language>
      </option>
    </options>
  </field>
        <field keyName="1_config.name" type="input">
            <label>
                <language value="en-US">Configuration name for ses-
sion</language>
            </label>
            <description>
                <language value="en-US">Name of session configura-
tion</language>
            </description>
        </field>
        <field keyName="1_com.servername" type="input">
            <label>
                <language value="en-US">Host address</language>
            </label>
            <description>
                <language value="en-US">Address of host</language>
            </description>
        </field>
        <field keyName="1_com.ibm5250User" type="input">
            <label>
```

```
                <language value="en-US">User ID</language>
            </label>
            <description>
                <language value="en-US">User name for Auto logon</lan-
guage>
            </description>
        </field>
        <field keyName="1_com.ibm5250SUBSPW" type="input">
            <label>
                <language value="en-US">Password</language>
            </label>
            <description>
                <language value="en-US">Password for Auto logon</lan-
guage>
            </description>
        </field>
        <field keyName="1_com.ibm5250UseAltLogin" type="checkbox">
            <label>
                <language value="en-US">Use Alternate Login</language>
            </label>
            <description>
                <language value="en-US">Check this to fill in the User
name and password in the Sign-on screen</language>
            </description>
        </field>
        <field keyName="1_com.tn5250maindevice" type="input">
            <label>
                <language value="en-US">Device name</language>
            </label>
            <description>
                <language value="en-US">Name of display</language>
            </description>
        </field>
        <field keyName="1_com.ibm5250Model" type="select">
            <label>
                <language value="en-US">5250 Model</language>
            </label>
            <description>
                <language value="en-US">0=3179-2, 1=3180-2, 2=3196-A1,
3=3477-FC, 4=3477-FG, 5=5251-11, 6=5291-1, 7=5292-2, 8=5555-C01,
9=5555-B01,10=3812-1, 11=5553-B01</language>
            </description>
            <options>
                <option selected="true" value="0">
                    <language value="en-US">3179-2 24x80 color</lan-
guage>
                </option>
                <option value="1">
                    <language value="en-US">3180-2 27x132 mono</lan-
guage>
                </option>
                <option value="2">
                    <language value="en-US">3196-A1 24x80 mono</lan-
guage>
                </option>
```

```
                <option value="3">
                    <language value="en-US">3477-FC 27x132 color</lan-
guage>
                </option>
                <option value="4">
                    <language value="en-US">3477-FG 27x132 mono</lan-
guage>
                </option>
                <option value="5">
                    <language value="en-US">5251-11 24x80 mono</lan-
guage>
                </option>
                <option value="6">
                    <language value="en-US">5291-1 24x80 mono</lan-
guage>
                </option>
                <option value="7">
                    <language value="en-US">5292-2 24x80 color</lan-
guage>
                </option>
                <option value="8">
                    <language value="en-US">5555-C01 24x80 Double-Byte
color</language>
                </option>
                <option value="9">
                    <language value="en-US">5555-B01 24x80 Double-
Byte</language>
                </option>
                <option value="10">
                    <language value="en-US">3812-1 Printer</language>
                </option>
                <option value="11">
                    <language value="en-US">5553-B01 Printer</lan-
guage>
                </option>
            </options>
        </field>
        <field keyName="1_com.useSSL" type="checkbox">
            <label>
                <language value="en-US">Use secure sockets -
SSL</language>
            </label>
            <description>
                <language value="en-US">Use a secure SSL/TLS connec-
tion to host</language>
            </description>
        </field>
        <field keyName="1_com.valServerCert" type="checkbox">
            <label>
                <language value="en-US">Authenticate server certifi-
cate</language>
            </label>
            <description>
                <language value="en-US">Verify that the server certif-
icate is from a Trusted root certificate authority</language>
            </description>
```

```
        </field>
        <field keyName="1_emu.ibmCodePage" type="select">
            <label>
                <language value="en-US">Host Code-Page</language>
            </label>
            <description>
                <language value="en-US">EBCDIC Codepage</language>
            </description>
            <options>
                <option value="037">
                    <language value="en-US">037 AU,CA, NZ, PT, ZA,
US</language>
                </option>
                <option value="1140">
                    <language value="en-US">1140 AU,CA, NZ, PT, ZA,
US</language>
                </option>
                <option value="273">
                    <language value="en-US">273 AT, DE</language>
                </option>
                <option value="1141">
                    <language value="en-US">1141 AT, DE</language>
                </option>
                <option value="1388">
                    <language value="en-US">1388 CN</language>
                </option>
                <option value="880">
                    <language value="en-US">1025 Cyrillic</language>
                </option>
                <option value="1025">
                    <language value="en-US">1025 Cyrillic</language>
                </option>
                <option value="277">
                    <language value="en-US">277 DK, NO</language>
                </option>
                <option value="1142">
                    <language value="en-US">1142 DK, NO</language>
                </option>
                <option value="278">
                    <language value="en-US">278 FI, SE</language>
                </option>
                <option value="1143">
                    <language value="en-US">1143 FI, SE</language>
                </option>
                <option value="875">
                    <language value="en-US">875 GR</language>
                </option>
                <option value="280">
                    <language value="en-US">280 IT</language>
                </option>
                <option value="1144">
                    <language value="en-US">1144 IT</language>
                </option>
                <option value="930">
                    <language value="en-US">939 JP</language>
```

```
                    </option>
                    <option value="930E">
                        <language value="en-US">939 JP</language>
                    </option>
                    <option value="939">
                        <language value="en-US">939 JP</language>
                    </option>
                    <option value="1399">
                        <language value="en-US">1399 JP</language>
                    </option>
                    <option value="933">
                        <language value="en-US">933 KR</language>
                    </option>
                    <option value="1364">
                        <language value="en-US">1364 KR</language>
                    </option>
                    <option value="870">
                        <language value="en-US">870 Latin 2</language>
                    </option>
                    <option value="284">
                        <language value="en-US">284 ES, Latin Amer-
ica</language>
                    </option>
                    <option value="1145">
                        <language value="en-US">1145 ES, Latin Amer-
ica</language>
                    </option>
                    <option value="285">
                        <language value="en-US">285 GB</language>
                    </option>
                    <option value="1146">
                        <language value="en-US">1146 GB</language>
                    </option>
                    <option value="297">
                        <language value="en-US">297 FR</language>
                    </option>
                    <option value="1147">
                        <language value="en-US">1147 FR</language>
                    </option>
                    <option selected="true" value="500">
                        <language value="en-US">500 International</lan-
guage>
                    </option>
                    <option value="1148">
                        <language value="en-
US">1148International</language>
                    </option>
                    <option value="1149">
                        <language value="en-US">1149 IS</language>
                    </option>
                    <option value="1026">
                        <language value="en-US">1026 TR</language>
                    </option>
                    <option value="1155">
                        <language value="en-US">1155 TR, Euro</language>
                    </option>
```

```
            <option value="937">
                <language value="en-US">937 TW</language>
            </option>
            <option value="1371">
                <language value="en-US">1371 TW</language>
            </option>
        </options>
    </field>
<field keyName="1_emu.anyCmdResets" type="checkbox">
        <label>
          <language value="en-US">Any command key resets error</lan-
guage>
        </label>
        <description>
          <language value="en-US">If an error occurred, normally the
keyboard is unlocked with the Reset command key only. If this option
is checked, any command key reset the error state and unlock the key-
board</language>
        </description>
    </field>
    <field keyName="1_emu.errToStatBar" type="checkbox">
        <label>
          <language value="en-US">Display error on status bar</lan-
guage>
        </label>
        <description>
          <language value="en-US">Display error on status bar in
stead of last line of emulation</language>
        </description>
    </field>
    <field keyName="1_emu.anywhereOk" type="checkbox">
        <label>
          <language value="en-US">Allow cursor in protected
area</language>
        </label>
        <description>
          <language value="en-US">Normally you cannot put the cursor
outside an input field. With this option checked, the cursor is moved
to the pointed position in the protected area</language>
        </description>
    </field>
</presentation>

</managedAppConfiguration>
```

MDM Parameter list

General parameters			
Version number for the configuration	config.version	number	Integer
Web Address where Glink can download config file	config.url	http/https address to config file	String
Name of new or existing host session	config.name	Define name for a session or GeneralOptions	String
Remove a session, General options or everything	config.remove	Name of session to remove or GeneralOptions / all	String
Send activation key to the devices	config.actkey	Product activation key with hyphens ('-') removed	String
Remove all toolbar definitions	toolbar.remove	true, false	Boolean
Send license	config.lic	License info with \n as line-separator	String
General options			
Screen orientation - Android only	screen.orientation	**0=Any**, 1=Portrait, 2=Landscape	Integer
Auto-connect session at start up	screen.autoConnect	true, **false**	Boolean
Auto-reconnect if connection lost - Android only	screen.autoReconnect	true, **false**	String
Disconnect at lock screen - Android only	screen.discntAtLock	true, **false**	Boolean
Never Auto-Lock when connected	screen.noAutoLock	true, **false**	Boolean
Ignoring battery optimizations - Android only	screen.ignoreBattery	true, **false**	Boolean
Keyboard up	screen.showKeyboard	**0=Default**, 1=On, 2=Off	Integer
Toolbar - click on keypress	screen.toolbarClicks	**0=Off**, 1=Sound 1, 2=Sound 2	Integer
Toolbar - vibrate on keypress in millisec - Android only	screen.toolbarVibr	**0=Off**, 20, 50, 100, 150, 200, 300	Integer
Toolbar - popup on keypress - Android only	screen.toolbarPopup	**0=Default**, 1=On, 2=Off	Integer
Send notification sound	screen.sndSound	**-1 (Off)=Default**, 0-98	Integer
Receive notification sound	screen.rcvSound	**-1 (Off)=Default**, 0-98	Integer

Show Sessions view when disconnected - Android only	screen.showSess	true, **false**	Boolean
Alert sound - Android only	screen.alertSound	0=Off, 1=**Android notification**, 2=Android alarm	Integer
Language	screen.language	Default, **us=US English**, de=DE German, pt=PT Portuguese, fr=FR French, no=NO Norwegian	String
Password	screen.cfgPassword		String
Key repeat – iOS Only	Screen.keyRepeat	true, **false**	Boolean
Show IOS status bar - iOS Only	screen.iosStatusBar	**0=Always**, 1=Portrait only, 2=Never	Integer
Show Menu/Action bar items - Android only			
On Menu (by default)			
Sessions	screen.mitem8	0= **Menu**, 1=Action bar, 2=Hide	Integer
Settings	screen.mitem10	0= **Menu**, 1=Action bar, 2=Hide	Integer
Scrollback	screen.mitem11	0= **Menu**, 1=Action bar, 2=Hide	Integer
Print screen	screen.mitem12	0= **Menu**, 2=Hide	Integer
Record macro	screen.mitem13	0= **Menu**, 2=Hide	Integer
Debug mode	screen.mitem14	0= **Menu**, 2=Hide	Integer
Print logging	screen.mitem15	0= **Menu**, 2=Hide	Integer
New Glink Window	screen.mitem21	0= **Menu**, 1=Action bar, 2=Hide	Integer
Remove Glink Window	screen.mitem22	0= **Menu**, 1=Action bar, 2=Hide	Integer
General options	screen.mitem16	0= **Menu**, 1=Action bar, 2=Hide	Integer
Export configurations	screen.mitem23	0= **Menu**, 2=Hide	Integer
Help	screen.mitem18	0= **Menu**, 1=Action bar, 2=Hide	Integer
About	screen.mitem19	0= **Menu**, 2=Hide	Integer
Exit	screen.mitem25	0= **Menu**, 2=Hide	Integer
On Action bar (by default)			

Distributing Glink configurations from an EMM/MDM

Toggle toolbar	screen.mitem24	1=**Action bar**, 2=Hide	Integer
Toggle keyboard	screen.mitem1	1=**Action bar**, 2=Hide	Integer
Connect	screen.mitem6	0= Menu, 1=**Action bar**, 2=Hide	Integer
Disconnect	screen.mitem9	0= Menu, 1=**Action bar**, 2=Hide	Integer
Enter	screen.mitem7	1=**Action bar**, 2=Hide	Integer
Pop up on Action bar			
Scan barcode	screen.mitem2	0= Menu, 1=**Action bar**	Integer
Print data	screen.mitem3	0= Menu, 1=**Action bar**	Integer
Debug data	screen.mitem4	0= Menu, 1=**Action bar**	Integer
SSL/SSH info	screen.mitem17	1=**Action bar**, 2=Hide	Integer
Select Glink	screen.mitem20	0= Menu, 1=**Action bar**	Integer
Pause	screen.mitem26	0= Menu, 1=**Action bar**	Integer

Communication			
Comms interface	com.type	Telnet, TN5250, TN3270, SSH, Ggate, TNVIP, Raw	String
Telnet options			
Server address	com.serverName		String
Terminal type	com.telnetType	**vt220**	String
Line terminator	com.telnetCRNUL	**0=CR+LF**, 1=CR+NULL, 2=CR	Integer
Use IP for BRK	com.telnetUseIP	true, **false**	Boolean
Binary mode	com.telnetBinary	true, **false**	Boolean
Simulate parity	com.telnetSimulateParity	true, **false**	Boolean
Keep alive type	com.keepAtype	**0=TCP**, 1=Telnet	Integer
Keep alive interval	com.keepAlive		String
Wait for login prompt	com.tnWaitUsr		String
Login with	com.tnUser		String
Wait for password prompt	com.tnWaitPsw		String

Password	com.tnPassword		String
Wait for command	com.waitprompt		String
Do command	com.docommand		String
Use secure sockets - SSL	com.useSSL	true, **false**	Boolean
Authenticate server certificate	com.valServerCert	true, **false**	Boolean
File name client certificate:	com.clientCert		String
Password client certificate:	com.clientCertPW		String
Use SSH connection	com.useSSH	true, **false**	Boolean
Server address:	com.sshHost		String
User name:	com.sshUser		String
Password:	com.sshPassword		String
Private key file:	com.sshKeyFile		String
Keep alive interval in seconds	com.sshKeepA	**0**	Integer
SSH options			
Server address	com.serverName		String
Use SSH connection	com.useSSH	**true**	Boolean
User name:	com.sshUser		String
Password:	com.sshPassword		String
Private key file:	com.sshKeyFile		String
Terminal type:	com.sshTerminal	**vt220**	String
Number of cols:	com.sshCols	**80**	Integer
Number of rows:	com.sshRows	**24**	Integer
Keep alive interval in seconds	com.sshKeepA	**0**	Integer
Wait for command	com.waitprompt		String
Do command	com.docommand		String
TN3270 options			
Server address	com.serverName		String

Distributing Glink configurations from an EMM/MDM

3270 model	com.ibm3270Model	**0=3279-2**, 1=3279-3, 2=3278-1, 3=3278-2, 4=3278-3, 5=3278-4, 6=3278-5, 7=3287-1, 8=3279-2E, 9=3279-3E, 10=3278-1E, 11=3278-2E, 12=3278-3E, 13=3278-4E, 14=3278-5E	Integer
LU name	com.tn3270LuName		String
Use Extended TN3270	com.tn3270Extended	**true**, false	Boolean
Send Associated LU	com.tn3270AssociateLU	true, **false**	Boolean
Keep alive type	com.keepAtype	**0=TCP**, 1=Telnet	Integer
Keep alive interval	com.keepAlive		String
Use secure sockets - SSL	com.useSSL	true, **false**	Boolean
Authenticate server certificate	com.valServerCert	true, **false**	Boolean
File name client certificate:	com.clientCert		String
Password client certificate:	com.clientCertPW		String
Login with	com.tnUser		String
Password	com.tnPassword		String
Use external validation app - Android only	com.validateApp	true, **false**	Boolean
TN5250 options			
Server address	com.serverName		String
5250 model	com.ibm5250Model	**0=3179-2**, 1=3180-2, 2=3196-A1, 3=3477-FC, 4=3477-FG, 5=5251-11, 6=5291-1, 7=5292-2, 8=5555-C01, 9=5555-B01, 10=3812-1, 11=5553-B01	Integer
Device Name	com.tn5250MainDevice		String
Keep alive type	com.keepAtype	**0=TCP**, 1=Telnet	Integer
Keep alive interval	com.keepAlive		String
User ID	com.ibm5250User		String
Password	com.ibm5250SUBSPW		String
Use alternate auto login	com.ibm5250UseAltLogin	true, **false**	Boolean

Use secure sockets - SSL	com.useSSL	true, **false**	Boolean
Authenticate server certificate	com.valServerCert	true, **false**	Boolean
File name client certificate:	com.clientCert		String
Password client certificate:	com.clientCertPW		String
Message queue	com.ibm5250IBMMSGQNAME	**QSYSOPR**	String
Message library	com.ibm5250IBMMSGQLIB	***LIBL**	String
Font ID	com.ibm5250IBMFONT	**11**	Integer
Form feed type	com.ibm5250IBMFORMFEED	**A=Auto feed sheets**, C=Continuous paper, U=Manual feed sheets	String
TNVIP options			
Server address	com.serverName		String
CoName or Resource	com.serverTarget		String
Terminal type	com.telnetType		String
Initial turn	com.initialTurn	**0=Default**, 1=Glink, 2=Host	Integer
Keep alive type	com.keepAtype	**0=TCP**, 1=Telnet	Integer
Keep alive interval	com.keepAlive		String
Use secure sockets - SSL	com.useSSL	true, **false**	Boolean
Authenticate server certificate	com.valServerCert	true, **false**	Boolean
Ggate options			
Server address	com.serverName		String
Alternate server address	com.altServerName		String
Random connect	com.tcpConnectRandom	true, **false**	Boolean
Use secure sockets - SSL	com.useSSL	true, **false**	Boolean
Authenticate server certificate	com.valServerCert	true, **false**	Boolean
CoName or Resource	com.serverTarget		String
Ggate protocol	com.ggateType	**DSA**, DIWS	String
Remote mailbox name	com.ggateDa		String

Distributing Glink configurations from an EMM/MDM

Remote mailbox extension	com.ggateDx		String
Remote node name	com.ggateDn		String
Local mailbox name	com.ggateMn		String
User ID	com.ggateDu		String
Password	com.ggatePw		String
Project	com.ggateDp		String
Billing	com.ggateDb		String
Host mode	com.ggateHm	**CXI**, DN, DPS7, DPS8, IBM, ROUTER, UNIX	String
Terminal mode	com.ggateTm	See Glink	String
GRTS/LID user string	com.ggateUr		String
IBM log mode	com.ggateLm		String
Additional parameters	com.ggateExtra		String
Keep alive (secs)	com.ggateKeepAliveInt	**0**	Integer
Initial turn	com.initialTurn	**0=Default**, 1=Glink, 2=Host	Integer
Send commands as data	com.ggateCommandsAsData	true, **false**	Boolean
Raw TCP/IP options			
Server address	com.serverName		String
Use SSH connection	com.useSSH	true, **false**	Boolean
User name:	com.sshUser		String
Password:	com.sshPassword		String
Terminal type:	com.sshTerminal	**vt220**	String
Number of cols:	com.sshCols	**80**	Integer
Number of rows:	com.sshRows	**24**	Integer
Wait for command	com.waitprompt		String
Do command	com.docommand		String
Proxy server options			
Use server	com.useProxy	true, **false**	Boolean
Address	com.proxyN		String
Port	com.proxyP	**30855**	Integer

Secure communication	com.proxyEncrypt	true, **false**	Boolean
User disconnect	com.proxyDC	**0**	Integer
Connection lost	com.proxyDH	**300**	Integer
Emulation			
Emulation Mode	emu.type	3270, 5250, VT, ANSI, VIP, DKU, TTY, 7102, 7700	String
VT options			
ECHO mode	emu.vtInitEcho	**true**, false	Boolean
ROLL mode	emu.vtInitRoll	**true**, false	Boolean
Line mode	emu.vtLineMode	**0=Off**, 1=Basic, 2=Extended	Integer
Add LFs to CRs	emu.vtAddLFs	true, **false**	Boolean
No column 81	emu.vtNo81	true, **false**	Boolean
Backspace key action	emu.vtDelToBs	**0=BS**, 1=DEL, 2=BS+DEL	Integer
Answerback string	emu.vtAnswerB		String
VT DA Alias	emu.vtAliasDA	0=Glink,1=VT100,2=VT101,3=VT102,4=VT125,5=VT131,6=VT220,7=VT240,8=VT320,9=VT340,10=VT420,**11=VT440**	Integer
F5 send PuTTY sequence	emu.vtAltF5	true, **false**	Boolean
Mute host alarm	emu.muteHAlarm	true, **false**	Boolean
ANSI options			
ECHO mode	emu.vtInitEcho	**true**, false	Boolean
ROLL mode	emu.vtInitRoll	**true**, false	Boolean
Line mode	emu.vtLineMode	**0=Off**, 1=Basic, 2=Extended	Integer
Add LFs to CRs	emu.vtAddLFs	true, **false**	Boolean
No column 81	emu.vtNo81	true, **false**	Boolean
Backspace key action	emu.vtDelToBs	**0=BS**, 1=DEL, 2=BS+DEL	Boolean
Answerback string	emu.vtAnswerB	ansi	String

Distributing Glink configurations from an EMM/MDM

VT / ANSI / VIP 7800 / VIP 7700 / DKU 7107 / DKU 7102 Transliteration			
UTF-8 encoding	emu.UTF8Encoding	true, **false**	Boolean
8-bit Host	emu.host8bit	**true**, false	Boolean
Allow lowercase to host	emu.allowLcase	**true**, false	Boolean
Host character set	emu.charset	**Default (Latin 1)**, ISO-8859-2 Latin 2, ISO-8859-5 Latin/Cyrillic, IBM-437, IBM-850, Windows-1251, Windows-1253, Windows-1257, koi8-r	String
National transliteration (7-bit host)	emu.xlitFile8	**Default (No transliteration)**, CYR Cyrillic, FIN Finnish, FRA French, FRC French-Canadian, GER German, GR Greek, ITA Italian, LT2 Latin 2, NOR Norwegian, SF Swedish/Finnish, SPA Spanish, SWE Swedish, SWI Swiss, UK United Kingdom, US United States	String
Use SISO encodoing	emu.useSISO	true, **false**	Boolean
VT / ANSI / VIP 7800 / VIP 7700 / DKU 7107 / DKU 7102 General			
Destructive BS	emu.destructBS	**true**, false	Boolean
Capture on CR	emu.captureOnCR	**0=LF**, 1=CR, 2=CRLF	Integer
Scrollback length (in pages)	emu.sbackMaxPages	**32**	Integer
Initial screen width	emu.initColumns	40, **80**, 132	Integer
Initial screen height (rows)	emu.initRows	**24**	Integer
5250 options			
Any command key resets error	emu.anyCmdResets	**true**, false	Boolean

Display errors on status bar	emu.errToStatBar	**true**, false	Boolean
Treat line as error message line	emu.ibmErrLine	**0=OFF**, 1 -> 27	Integer
- If error message line contains	emu.ibmErrMsg		String
Mute host alarm	emu.muteHAlarm	true, **false**	Boolean
Mute key input alarm	emu.muteKAlarm	true, **false**	Boolean
Show SO/SI characters	emu.ibmShowSoSi	true, **false**	Boolean
Allow cursor in protected area	emu.anywhereOk	true, **false**	Boolean
Alternate arrow keys mode	emu.altArrowMode	true, **false**	Boolean
Show light pen fields in blue	emu.ibmLightPen	true, **false**	Boolean
Auto tab when input field is filled	emu.autoTabL	**true**, false	Boolean
Auto tab when field is filled with scanned data	emu.autoTabScan	**true**, false	Boolean
Continuous backspace	emu.continuousBS	true, **false**	Boolean
Bypass Right Adjust field check	emu.bypassRAcheck	true, **false**	Boolean
Only scanner input allowed in selector pen field	emu.onlyScanData	true, **false**	Boolean
Unicode data stream	emu.ibmUnicode	true, **false**	Boolean
3270 options			
Numeric checking	emu.numCheck3270	**0=None**, 1=Strict, 2=Emulator, 3=Relaxed	Integer
Treat line as error message line	emu.ibmErrLine	**0=OFF**, 1 -> 43	Integer
Mute host alarm	emu.muteHAlarm	true, **false**	Boolean
Mute key input alarm	emu.muteKAlarm	true, **false**	Boolean
Allow cursor in protected area	emu.anywhereOk	true, **false**	Boolean
IBM 5250 / IBM 3270 Host code page			
Host code page	emu.ibmCodePage	**500 International**, See list in Glink	String
IBM 5250 / IBM 3270 General			

Auto tab when input field is filled	emu.autoTabL	**true**, false	Boolean
Auto Tab host	emu.autoTabH	true, **false**	Boolean
Destructive BS	emu.destructBS	**true**, false	Boolean
Auto scroll to cursor if not visible	emu.aScrollToCu	**true**, false	Boolean
Scrollback length (in pages)	emu.sbackMaxPages	32	Integer
VIP 7800 options			
Auto tab when input field is filled	emu.autoTabL	**true**, false	Boolean
Auto Tab host	emu.autoTabH	true, **false**	Boolean
Start in TEXT mode	emu.vipInitText	**true**, false	Boolean
Extended status	emu.vipExtStatus	true, **false**	Boolean
Non-linear forms	emu.vipScrewyForms	true, **false**	Boolean
Preserve roll mode	emu.vipStickyRoll	**true**, false	Boolean
Suppress status line	emu.vipNoStatusLine	true, **false**	Boolean
TX on return	emu.vipTxRetMode	true, **false**	Boolean
ECHO mode	emu.vipInitEcho	true, **false**	Boolean
ROLL mode	emu.vipInitRoll	**true**, false	Boolean
Block mode	emu.vipInitBlock	true, **false**	Boolean
Disallow status line lock	emu.vipSuppressLock	true, **false**	Boolean
Space suppression	emu.vipSpaceSuppress	**true**, false	Boolean
Add CRLF to transmit	emu.vipAddCRLFs	true, **false**	Boolean
Add LFs to CRs	emu.vipAddLFs	true, **false**	Boolean
Send NL after XMIT	emu.vipNLAfterXmit	true, **false**	Boolean
Send ETX for end of transmission	emu.etxSend	**true**, false	Boolean
Enquiry reply string	emu.vipEnqString		String
7200 attributes	emu.vipV7200A	**0=No**, 1=Yes, 2=Extended	Integer
VIP 7700 options			
Auto tab when input field is filled	emu.autoTabL	**true**, false	Boolean
Auto Tab host	emu.autoTabH	true, **false**	Boolean

Start in TEXT mode	emu.vipInitText	**true**, false	Boolean
Wraparound tabbing	emu.dkuWrapTab	**true**, false	Boolean
Add LFs to CRs	emu.vipAddLFs	true, **false**	Boolean
Space suppression	emu.vipSpaceSuppress	**true**, false	Boolean
Suppress status line	emu.vipNoStatusLine	true, **false**	Boolean
ROLL mode	emu.vipInitRoll	true, **false**	Boolean
Show 7700 spaces	emu.vip77SpaceReplace	true, **false**	Boolean
VIP compatibility	emu.vipCompat	**0=Normal**, 1=ITTCourier, 2=ThomasBox	Integer
DKU 71707 options			
Auto tab when input field is filled	emu.autoTabL	**true**, false	Boolean
Blink/blank with ^/~	emu.dkuBlinkAv	**true**, false	Boolean
Show ^/~ in blink/blank	emu.dkuBlinkShow	**true**, false	Boolean
Cursor straight up/down	emu.dkucud	true, **false**	Boolean
Allow cursor out of field	emu.dkucuf	**true**, false	Boolean
New line after Xmit	emu.dkuNewLine	**true**, false	Boolean
SDP attributes	emu.dkuSDPAtt	true, **false**	Boolean
Wrap on page overflow	emu.dkuWrap	true, **false**	Boolean
Wraparound tabbing	emu.dkuWrapTab	**true**, false	Boolean
ROLL mode	emu.dkuInitRoll	true, **false**	Boolean
Use SS2 for 8bit chars	emu.dkuExtend	true, **false**	Boolean
Send ETX for end of transmission	emu.etxSend	**true**, false	Boolean
Wincom compatibility	emu.dkuWincom	true, **false**	Boolean
Fix DKU attributes	emu.dkuFixAttr	**true**, false	Boolean
DKU model	emu.dkuTermID	**0=7107**, 1=7211	Integer
Right justify with	emu.dkuRjfChar	**32=Space**, 48=Zero, 42= Asterisk, 36=Dollar, 164=Euro, 250=Pound	Integer
DKU parameters not configurable in Glink			

Printer lines per page	emu.dkuPrtNbl		Integer
Printer columns	emu.dkuPrtNbc		Integer
Printer cps	emu.dkuPrtCps		Integer
Enable local forms	emu.formsEnable		Boolean
DKU 7102 options			
Add LFs to CRs	emu.vipAddLFs	true, **false**	Boolean
ECHO mode	emu.dkuInitEcho	true, **false**	Boolean
ROLL mode	emu.dkuInitRoll	**true**, false	Boolean
Tab key sends HT	emu.dkuTabSendsHT	true, **false**	Boolean
No column 81	emu.vtNo81	true, **false**	Boolean
Screen			
Toolbar options			
Button width with scaling factor	toolbar.buttonScaleX	**1.0**	String
Button height with scaling factor	toolbar.buttonScaleY	**1.0**	String
Button font scaling factor	toolbar.buttonScaleF	**1.0**	String
Max vertical toolbar columns	toolbar.vtbNum	**Default**, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	String
Max visual toolbar columns	toolbar.vtbVis	**Default**, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	String
Max visual buttons per row	toolbar.htbVisCol	**Default**, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	String
Max visual toolbar rows	toolbar.htbNumRow	**Default**, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	String
Toolbar buttons			
Toolbar button on row L position N	toolbarL.cmdN	Create Toolbar config in Glink, export and copy	String
Remove all toolbar definitions	toolbar.remove	true, false	Boolean

Screen options			
Font size	screen.fontSize	-1=Auto, -2=Fit screen, 6-36	Integer
Font name	screen.fontName	**Android: Default**, Courier-New, Courier-NewBold, Cousine-Regular, Cousine-Bold, DroidSansMono, Menlo-Regular, Menlo-Bold, Source-CodePro-Regular, SourceCodePro-Semi Bold, SourceCodePro-Bold **iOS: Couri-erNew**, Courier, Menlo-Regular	String
Font style - iOS Only	screen.fontStyle	0=Plain, **1=Bold**, 2=Italic, 3=Bold & Italic	Integer
Cursor type	screen.cursortype	0=Block, **1=Bar**, 2=Line	Integer
Cursor color	screen.usecursorcolor	**0=Default**, 1=User defined (white ch), 2= User defined (black ch)	Integer
Cursor color	screen.cursorColor	Change color in Glink and export config	String
3D variable fields	screen.variable3Ds	**0=Off (5250/3270)**, **1=On without attribute(VT/ANSI/VIP/DKU/7700/7102)**, 2=On with attribute, 3=On without underline, 4=Show as underline	Integer
Blinking cursor	screen.cursorBlink	true, **false**	Boolean
White background 3D fields	screen.whiteBg3D	true, **false**	Boolean
Show toolbar	screen.multiLineTB	**0=Auto,** 1=Multiline, 2=Single-line, 3=Off, 4=Vertical	Integer
Limit screen view (x,y)	screen.limitXY	col,row	String
Use double tap to	screen.dblTapEnt	**0=Reset screen size**, 1=Perform Enter, 2=Toggle Action bar On/Off (On), 3=Toggle Action bar On/Off (Off), 4=Enable/Disable touchscreen (Enabled) **Android only**, 5==Enable/Disable touchscreen (Disabled) **Android only**	Integer

Distributing Glink configurations from an EMM/MDM

Double tap again to confirm - Android only	screen.dbltapcfirm	true, **false**	Boolean
Extra line spacing	screen.extraPixelsY	From -4 to +8 **Default: 0**	Integer
Force fixed character width	screen.variableFont	true, **false**	Boolean
Disable zoom	screen.noZoom	true, **false**	Boolean
Show key input on status line	screen.showkeyinput	true, **false**	Boolean
Function keys	screen.hotspotFn	**true**, false	Boolean
Option numbers ("nn." or "nn=")	screen.hotspotNn	0=Off, 1=On, **2=On with Enter**	Integer
Option numbers (" nn ")	screen.hotspotN2	**0=Off**, 1= On, 2=On with Enter	Integer
Option characters ("aa=")	screen.hotspotAa	**0=Off**, 1= On, 2=On with Enter	Integer
URL address	screen.hotspotUrl	0=Off, 1=Open in browser, **2=Open in Glink**	Integer
Screen colors			
Color selection			
Screen color	screen.scrColor	Make Color selection in Glink and export config	Integer
Status line color	screen.stsColor	Make Color selection in Glink and export config	Integer
Color Adjustment			
Dark forground			
Black	screen.foreground_BlackLo	Make Color adjustment in Glink and export config	String
Blue	screen.foregroundBlueLo	Make Color adjustment in Glink and export config	String
Green	screen.foregroundGreenLo	Make Color adjustment in Glink and export config	String
Cyan	screen.foregroundCyanLo	Make Color adjustment in Glink and export config	String
Red	screen.foregroundRedLo	Make Color adjustment in Glink and export config	String

Magenta	screen.foregroundMagentaLo	Make Color adjustment in Glink and export config	String
Yellow	screen.foregroundYellowLo	Make Color adjustment in Glink and export config	String
White	screen.foregroundWhiteLo	Make Color adjustment in Glink and export config	String
Bright forground			
Black	screen.foreground_BlackHi	Make Color adjustment in Glink and export config	String
Blue	screen.foregroundBlueHi	Make Color adjustment in Glink and export config	String
Green	screen.foregroundGreenHi	Make Color adjustment in Glink and export config	String
Cyan	screen.foregroundCyanHi	Make Color adjustment in Glink and export config	String
Red	screen.foregroundRedHi	Make Color adjustment in Glink and export config	String
Magenta	screen.foregroundMagentaHi	Make Color adjustment in Glink and export config	String
Yellow	screen.foregroundYellowHi	Make Color adjustment in Glink and export config	String
White	screen.foregroundWhiteHi	Make Color adjustment in Glink and export config	String
Background			
Black	screen.backgroundBlack	Make Color adjustment in Glink and export config	String
Blue	screen.backgroundBlue	Make Color adjustment in Glink and export config	String
Green	screen.backgroundGreen	Make Color adjustment in Glink and export config	String
Cyan	screen.backgroundCyan	Make Color adjustment in Glink and export config	String

Distributing Glink configurations from an EMM/MDM

Red	screen.backgroundRed	Make Color adjustment in Glink and export config	String
Magenta	screen.backgroundMagenta	Make Color adjustment in Glink and export config	String
Yellow	screen.backgroundYellow	Make Color adjustment in Glink and export config	String
White	screen.backgroundWhite	Make Color adjustment in Glink and export config	String
VT / ANSI / DKU / 7102 attribute mapping			
Color scheme	screen.sgrColor	**0=Mono**,1= 4-color A, 2=4-color B, 3=7-color	Integer
Use underline	screen.sgrUseUnderline	**true**, false	Boolean
Use blink	screen.sgrUseBlink	**true**, false	Boolean
Attributes to map	screen.sgrusermap	Create mapping in Glink and export config	String
VIP attribute mapping			
Parameter only needed when default is changed	screen.orattss	Create mapping in Glink and export config	
Parameter only needed when default is changed	screen.andattss	Create mapping in Glink and export config	
Devices			
Print setup			
Printer type - Android only	print.type	**File=Default**, LPD=LPD server, BT=Bluetooth printer	String
Font size - iOS only		**14**	
Page Orientation - iOS only		**Portrait**	
Printer timeout (secs)	print.timeout	**5**	Integer
LPD server address - Android only	print.lpdServerName		String
LPD queue name - Android only	print.lpdQueueName		String
Pre-print character - Android only	print.pccBefore	**0=None**, 1=CR, 2=CR+LF, 3=CR+FF, 4=CR+VT	Integer

Glink for iOS User Guide *Gallagher & Robertson*

Post-print character - Android only	print.pccAfter	**0=None**, 1=CR, 2=CR+LF, 3=CR+FF, 4=CR+VT	Integer
Transparent print - Android only	print.transparent	true, **false**	Boolean
Use print ctl for host data - Android only	print.pCtlHostData	true, **false**	Boolean
Bluetooth device - Android only	print.btDevice		String
Init string - Android only	print.btInit		String
Character encoding - Android only	print.btEncoding	**IBM437**	String
DKU option			
Use 0x19 as SS2	print.ss2X19	true, **false**	Boolean
Use Esc E as SS2	print.ss2EscE	true, **false**	Boolean
Combine double escapes	print.ss2EscEsc	true, **false**	Boolean
Barcode reader setup			
Barcode reader device type	print.bcEnable	**Android:** 0=Keyboard Wedge, 7=Keyboard Wedge A, 1=Camera, 8=AML Scanner, 16=Chainway, 10=Cipherlab, 9= Datalogic, 4=Denso Scanner, 5=Honeywell Scanner, 11=M3 Mobile, 17=Movfast, 15=Newland, 12=Point Mobile, 14=Seuic, 6=Socket Mobile, 3=Unitech Scanner, 13=Urovo scanner, 2=Zebra skanner **iOS: 0=Keyboard Wedge**, 7=Keyboard Wedge A	Integer
Action after scan	print.bcDoAfter	**0=None**, 1=Tab, 2=Enter, 3=FldExit (5250)	Integer
Remove char(s) at start	print.bcRemFirst	**0**	Integer
Remove char(s) at end	print.bcRemLast	**0**	Integer
Add text before	print.bcAddFirst		String

Distributing Glink configurations from an EMM/MDM

Add text after	print.bcAddLast		String
Use keyboard mapping	print.bcUseKeyMap	**true**, false	Boolean
Show on status line - Android only	print.bcShow	**true**, false	Boolean
Profile name - Android only	print.dwProfile	**Glink**	String
Intent action - Android only	print.dwIntent	**no.gar.glink.data**	String
Intent string extra - Android only	print.dwIntentEx		String
Glink sets Output Mode (M3 Mobile only)	print.m3glinkmode	**true**, false	Boolean
Card data format string Linea/Infinea - iOS only	print.bcCardFmt	**{0}\\t{1}\\t{2}\\t{3}\\t{4}**	String
Keyboard			
Key input handling - Android only	keyboard.altInput	0=Auto, 1=Standard, 2=Extended, 3=Basic	Boolean
Map shift/F1.. to F13..	keyboard.F13F24	**true**, false	Boolean
Mappable modifier keys - Android only	keyboard.mapModifier	true, **false**	Boolean
Detect long press keys - Android only	keyboard.useLongPress	true, **false**	Boolean
Macro 1	keyboard.macroNN	Create macro in Glink, export config and copy	String
Mapped keys	keyboard.kc.NN_C	Create mapping in Glink, export config and copy	String

How to install SSL/TLS client certificates and SSH private key files

SSL/TLS Client certificates

How to import a PSKC #12 certificate (.p12 file)

If you have specified a file name for the client certificate in the Glink settings, Glink will check if a file with that name exists internally in the Glink file structure. If the file does not exist, Glink for iOS will display an error message and you must import it, see below. On Android, Glink will look for the file in the Glink auto import folder and import it if it exists. If not found, Glink gives the user the option to select a client certificate. If a file is selected, Glink will import the file and save it with the name specified in the Glink Settings regardless of the name of the file selected.

Distribute the p12 client certificate file to the auto-import folder (Android only)

Use MDM to distribute client certificates to the Glink auto import folder and Glink will pick it up as described above.

Use AppConfig parameter in MDM to download p12 client certificate (Android and iOS)

When configuring Glink from a MDM, you can use the config.url parameter to specify the URL to a p12 certificate file. If the file name is my_cert.p12 the certificate will be saved with that name and can be used from any session in the File name client certificate option.

Send p12 client certificate as e-mail attachment (iOS)

Another way to import the client certificate file (with extension *.glinki*) is to send an e-mail to your device with the file as an attachment.

Glink is set up to accept files with **.glinki** as the file name extension. So if your certificate is named *my_cert.p12*, rename it to *my_cert_p12.glinki*.

On your device, open your mail client, select the mail, and click on the attachment. Glink will then be launched and ask if you want to import the file as a new configuration (your configuration will not be overwritten in this case). The file will be stored internally in Glink as *my_cert_p12*. Use this name when you configure the certificate in Glink.

SSH private key file

How to import a private key file

Glink will handle files with *.glinki* as the file name extension, i.e *myfile.glinki*. One way to import the SSH private key file, is to send it as an e-mail to your device with *myfile.glinki* as an attachment. Click on the attachment and Glink will ask if you want to import it as a new configuration (your configuration will not be overwritten in this case).

The *myfile.glinki* must have the following format:

```
[file:my_pk_file]
Add the content of the private key here...
```

You can also append this at the end of your *config.glinki* file:

```
[file:my_pk_file]
Add the content of the private key here...
```
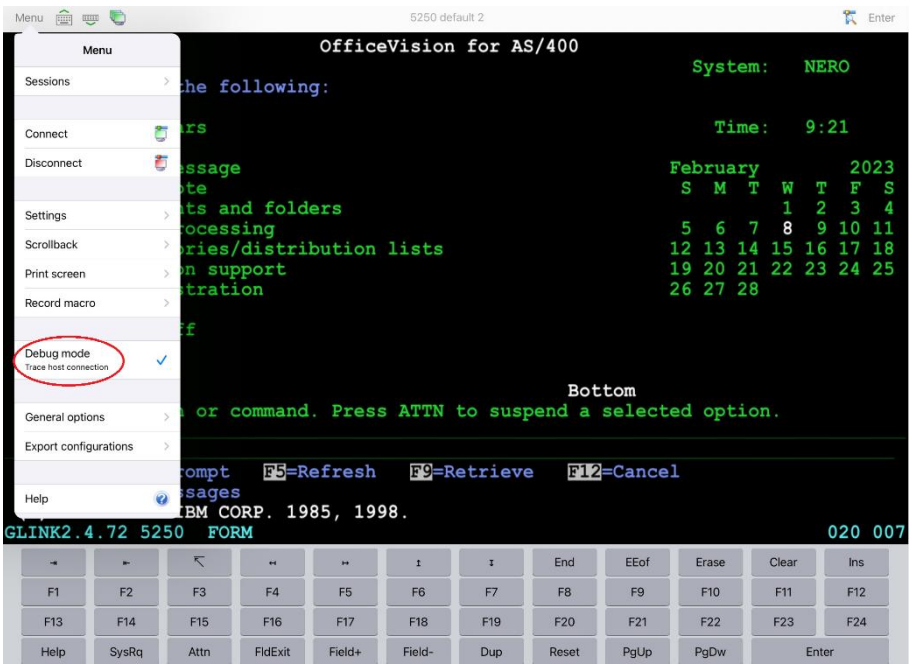
Glink will create a file with the file name *my_pk_file* with the content supplied.

In the Telnet or SSH configuration dialog box, enter *my_pk_file* in the *Private key file* field.
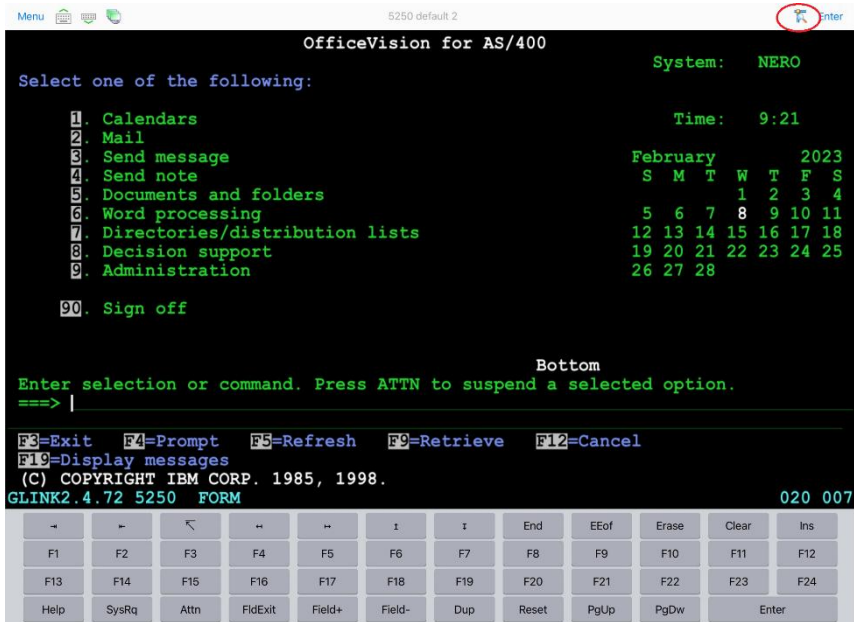
How to export/send a debug trace and configuration file

Enable, capture and export debug file
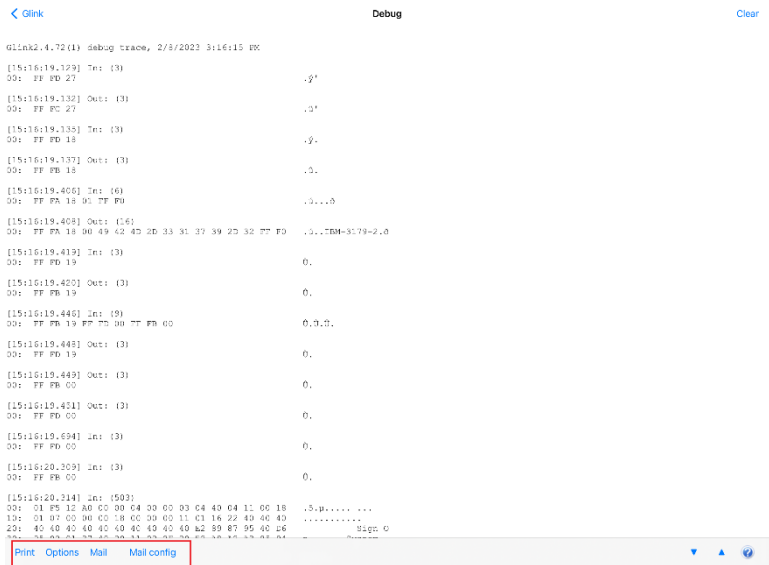
1. Enable debug mode: Tap **Menu -> Debug Mode**

2. Return to emulation screen and tap Disconnect + Connect
3. Interact with the host application until you have collected the request information

How to export/send a debug trace and configuration file

4.   Send debug trace file: tap **Debug icon -> Menu -> Mail** to send debug file as file attachment.

5. An alternative might be to print the debug file on an AirPrint printer: **Debug icon -> Menu -> Print**. Tap **Options** to set Print options like *Font size* and *Page orientation*.
6. From this menu you can also send the configuration file as an e-mail attachement: **Debug icon -> Menu -> Mail config**

Export configuration file

The config file (config.glinki) you can you send by e-mail from the device (**Menu -> Export configuration**) or from the Debug window (as described in the previous section).

The GlinkProxy Session Persistence Server

GlinkProxy runs on a Windows server, can be located close to the host computer, and is designed to extend the life of sessions between Glink terminal emulation clients and hosts. A device running Glink might go idle, enter power-saving mode or move out of WiFi range and prematurely terminate the session between Glink and the host. The *GlinkProxy* software maintains the session with the host, even if the device goes to sleep or the connection is temporarily lost. This enables Glink to seamlessly resume the session without loss of productivity when it reconnects to *GlinkProxy*.

How it works

GlinkProxy is a server application that acts as a link between a Glink terminal emulation client and a host, maintaining an established session with the host even though the client device might go idle or lose WiFi connection. This is how it works:

1. Glink connects to the GlinkProxy application server.

2. GlinkProxy connects to the host application and establishes a session with the host on behalf of Glink and keeps track of the session information.

3. GlinkProxy maintains the host connection, even if the connection between Glink and GlinkProxy goes idle or dead. You can configure how long GlinkProxy should maintain the host connection after the connection between Glink and GlinkProxy has been terminated. GlinkProxy can also differentiate between normal user initiated disconnects and unexpected disconnects with an error.

4. When a Glink client reconnects after having been idle or lost the connection, the GlinkProxy server sends the session information to Glink and Glink continues the session.

The following image provides a conceptual overview of GlinkProxy.
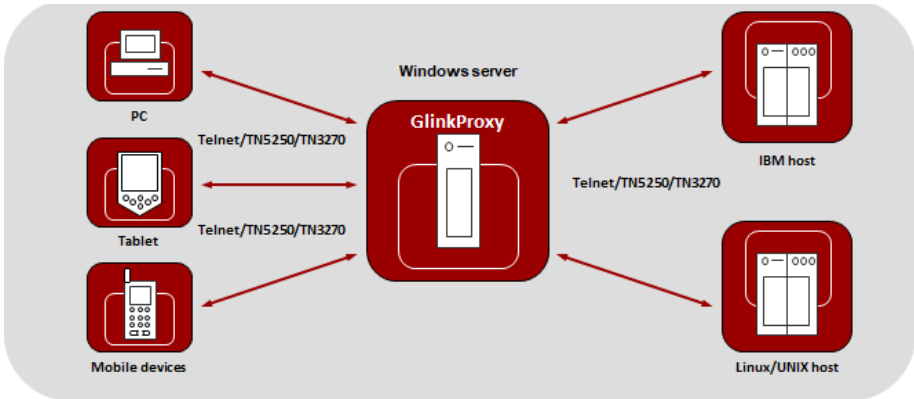
The GlinkProxy application runs on a Windows server that can be located near the host computer. It will allow the traffic from the mobile computers, tablets or PCs with Glink to be routed through the GlinkProxy server to the host. The GlinkProxy server will keep the connection to the host application even if the connection to the client computers with Glink is lost. When Glink reconnects, the session with the host application will resume as if the connection was not lost.

Delivery

The GlinkProxy application is delivered as a zip-file containing the GlinkProxy executable file and the product documentation.

Install and setup

Copy the file to the Windows server on which you want to run the GlinkProxy software.
Create a folder for GlinkProxy, for example C:\GlinkProxy, and unzip the GlinkProxy package here.
Start *glinkproxy.exe* without parameters by double-clicking the icon or typing in *glinkproxy.exe* at a command prompt. You will now be requested to enter your license information. Either type it in or copy-and-paste it into the text box. Click "OK" twice and "Allow access" to the network for GlinkProxy when Windows Defender Firewall or other firewall prompts for it.
You now have a running instance of GlinkProxy with default parameters.
With default parameters, GlinkProxy will

The GlinkProxy Session Persistence Server

Keep the host connection for 30 minutes if the Glink device goes idle, enters power-saving mode or moves out of WiFi range and prematurely terminates the session to GlinkProxy. If Glink reconnects within 30 minutes, the session is resumed as if it never was disconnected from GlinkProxy.

Disconnect the host connection immediately if the Glink user disconnects the session.

Allow Glink to change the disconnect timeout values.

Not send any keepalive packets unless specified by Glink.

Open the Settings menu if you want to change some of these values and Save the changes.

Below you can read more about the Settings menu, its parameters and how you can set them from the command prompt. You will also find a description of the proxy server parameters that can be set in Glink for iOS and Glink for Android.

Startup

If GlinkProxy is started without command line parameters, it will use default parameters the first time it is started and saved parameters on all following executions. If started with command line parameters (listed below), the command line parameters will override the previously saved parameters. For simplicity, it is recommended that you click Settings inside GlinkProxy to modify and save parameters and that you start GlinkProxy without any command line parameters.

Settings

Click "Settings" on the menu bar to open the settings menu.

In this menu you can configure:

Time to retain host connection when client disconnects in error. The server will preserve sessions to the host when the connection is terminated abnormally. This parameter may be used to specify a timeout (in seconds) after which the host connection will be closed (same as /ED command line parameter). Note that the Glink client may redefine this timeout if this has not been disabled (/ACR, /NCR or Allow client disconnect redefinition)

Time to retain host connection when client disconnects normally. This parameter specifies how a normal disconnect from the client should be processed. The default is zero, i.e., simply terminate the host connection immediately. Otherwise, the parameter tells the server to keep the connection active for the configured amount of time, allowing the client to reconnect until the timer expires (same as /ND command line parameter). Note that the Glink client may redefine this timeout if this has not been blocked (/ACR, /NCR or Allow client disconnect redefinition)

Time to keep host disconnect status available. This parameter specifies how long the server will retain host disconnect status (in seconds). The default value is 3600.

Use Telnet keepalive for host connection with interval. This parameter enables use of Telnet keepalives between GlinkProxy and the host. The default is not to use Telnet keepalives, but it can be very useful if the session is idle for long periods, and you want to avoid that you are disconnected by the host due to inactivity.

The GlinkProxy Session Persistence Server

Use TCP keepalive for host connection. This parameter enables use of TCP keepalives between GlinkProxy and the host. The default is not to use TCP keepalives. The actual keepalive interval is a Windows system parameter.

Allow client disconnect redefinition. This parameter allows redefinition of the /ND and /ED timeouts by the Glink client. This is the default, but you may use this parameter to override /NCR that might have been saved in Settings/Save.

Ask for confirmation when killing sessions. In the Manage sessions menu, you can kill selected sessions. When this option is enabled, the administrator must confirm the intention to kill the specific session.

Save Click to save modified parameter values
Restore Click to restore parameter values to the saved values
Defaults Click to set default parameter values
OK Click to save parameters and exit the settings menu
Cancel Click to cancel changes and exit the settings menu

Command line parameters

The GlinkProxy application is provided as a standalone program *glinkproxy.exe*. Its behavior is controlled by parameters set and saved in the Settings menu inside the application, but you can also use the following command line parameters to override the saved parameters.

Parameter	Description	Comment
/ACR	Allow Client Redefine	This parameter allows redefinition of the /ND and /ED timeouts by the Glink client. This is the default, but you may use this parameter to override /NCR that might have been saved in Settings/Save.
/DLT nn	Disconnect List Timeout	This specifies how long the server will retain host disconnect status (in seconds). The default value is 3600.
/DEBUG	Debug	If this is specified then a separate window will open to display debug information for server activity.
/ED nn	Error Disconnect timeout	The server will preserve sessions to the host when the connection is terminated abnormally or when the /ND parameter has kept the

		session in hold. The /ED parameter may be used to specify a timeout (in seconds) after which the host connection will be closed. Note that the Glink client may redefine this timeout if this has not been disabled by the /NCR command line parameter.
/EK	Enter Key	When this option is specified, the licensing form will be displayed to let you inspect or modify the licensing data. The server will terminate immediately after exiting the dialog
/KT nn	Telnet KeepAlive	This parameter enables sending of Telnet keepalive packets every nn seconds from GlinkProxy to the host. The default is not to send Telnet keepalive packets, but it can be very useful if the session is idle for long periods and you want to avoid being disconnected by the host due to inactivity.
/KA	TCP KeepAlive	This enables use of TCP keepalives between GlinkProxy and the host. The default is not to use TCP keepalives. The actual keepalive interval is a Windows system parameter.
/NCR	No Client Redefine	This parameter prevents redefinition of the /ND and /ED timeouts by the Glink client.
/ND nn	Normal Disconnect timeout	This specifies how a normal disconnect from the client should be processed. The default is zero, i.e., simply terminate the host connection immediately. Otherwise, the parameter tells the server to keep the connection active for that number of seconds, allowing the client to reconnect until the timer expires. Note that the Glink client may redefine this timeout if this has not been blocked by the /NCR command line parameter.
/PORT nn	Port number	The server listens by default on port number 30855 (0x7887). This parameter may be used to override this with a port number of your choice.

/VERBOSE	Verbose de-bug	Supplied as well as the /DEBUG parameter this tells the server to display all data sent and received (default is to display only the first 16 bytes of each block).

The Settings/Save option available while the server is running will save many of these settings and you will then only need to use the above to override the saved values. This applies to /ACR, /DLT, /ED, /KA, /NCR and /ND.

Glink for iOS/Android Proxy Server Settings

Glink 5250, Glink 3270, Glink VT and Glink for iOS/Android can communicate with TN5250, TN3270, Telnet, TNVIP and Ggate protocols via GlinkProxy to host systems. This is enabled in the **Menu -> Settings -> Proxy server** menu:

Parameter	Value	Description
Use server	on/OFF	Check this option to enable communication thru a GlinkProxy server
Address	ip-addr	Configure the address of the server running GlinkProxy (not the application host)
Port	nn	Configure the port number GlinkProxy is listening to. The default is 30855
User disconnect	nn	Specifies the number of seconds the proxy server should keep the connection to the host application after a user initiated disconnect. The default value is 0, which means that the connection to the host is disconnected immediately. This is what happens without a proxy server. Set this parameter to a non-zero value if you want that the proxy server should keep the session to the host the configured number of seconds after a user initiated disconnect

Con- nec- tion lost	nn	Specifies the number of seconds the proxy server should keep the connection to the host application after the connection to Glink is dropped with an error, for example if the device goes idle, enter power-saving mode, or move out of WiFi range. When the device is woken up or the network connection is reestablished, the user can reconnect and resume the session with the host application.

When Glink is configured to use a proxy server, it will send server address and keepalive settings to the proxy server. The proxy server will use these parameters to connect to the host application server and maintain the connection to it. Keepalive settings, user disconnect and connection lost settings from Glink will override settings in GlinkProxy if GlinkProxy is set up to allow Glink to change these parameters. For all communications, the proxy server is a transparent communication gateway between Glink and the host application.