



Glink VT

for

iOS

User Guide

<http://www.gar.no/>



Apple, iOS, iPad, iPhone and Mac are registered trademarks of Apple Inc.
Android, Chrome, Google, Google Play are trademarks of Google LLC.
Microsoft, Windows, MS, are registered trademarks of Microsoft Corp.
IBM and PC are registered trademarks of IBM Corp.

Any other product names are trademarks of their respective owners.

Glink VT for iOS, version 2.4.8
© Gallagher & Robertson AS, 2023
All Rights Reserved

GALLAGHER & ROBERTSON AS, Grini Næringspark 3, N-1361 Østerås, Norway
Tel: +47 23357800
www: <https://www.gar.no/>
e-mail: support@gar.no

Contents

Contents	i
Warranty	1
Privacy policy	2
Introduction	3
Features	3
Support	4
How to purchase/order a Glink license.....	4
Delivery and Installation	4
Setup.....	5
The Glink emulator screen	5
Action bar.....	5
Emulation screen.....	6
Toolbars	7
Soft/on-screen keyboard.....	7
Physical keys and keyboards.....	7
Getting started	8
Create a new session.....	8
Start a session	8
Switching session	9
Zoom and font size	9
Text size	9
Keyboard	10
External keyboard.....	10
Keyboard shortcuts.....	10
Shortcuts for keyboards with function keys	10
Shortcuts for keyboards without function keys	10
Other shortcuts	11
VT shortcuts	11
Keyboard mapping.....	11
Toolbar	12
Hotspots.....	12

Contents

Copy/Paste.....	12
Hyperlinks and mail addresses	13
Export and Import configurations.....	13
MDM configuration support.....	13
Proxy server.....	13
Gestures	14
Menu	15
Sessions	16
Connect.....	16
Disconnect.....	16
Settings	17
Scrollbar.....	17
Print screen.....	18
Record macro	18
Pause the macro.....	18
Wait for user input.....	19
Wait for a time interval.....	19
Debug mode	19
Debug.....	19
Print logging.....	20
General options.....	20
Never Auto-Lock when connected.....	21
Toolbar - Click on keypress	21
Show iOS status bar	21
Keyboard up.....	21
Auto-connect session at startup.....	21
Password - Protect Settings	22
Export configurations	23
Help.....	23
Connect icon (Green monitor).....	23
Keyboard icon	23
Toolbar icon	23
Settings - Communication.....	24
Telnet options	24
Server address	25
IPv6 address	25
Terminal type	26
Line terminator.....	26
Use IP (Interrupt Process) command for break	26
Binary mode.....	26
Keep alive	26

Keep alive type	26
Simulate parity	27
Telnet login	27
Wait for login prompt.....	27
Login with	28
Wait for password prompt	28
Password.....	28
Secure sockets	28
Use secure sockets - SSL.....	28
Authenticate server certificate	29
File name client certificate	29
Password client certificate	29
How to import a private key file.....	29
SSH Connection	30
Use SSH connection	30
Server address.....	30
User name.....	30
Password.....	31
Private key file.....	31
Keep alive interval.....	31
SSH options.....	32
Server address	32
User name	32
Password	32
Private key file	33
Terminal type	33
Number of cols.....	33
Number of rows.....	33
Keep alive interval	33
Settings – Emulation.....	34
VT / ANSI options	34
ECHO mode	34
ROLL mode	35
Line mode	35
Line mode options	35
Add LFs to CRs.....	36
No column 81.....	36
Backspace key action.....	36
Answerback string.....	36
VT DA Alias	36
F5 send PuTTY sequence.....	37
VT / ANSI - Transliteration	37
UTF-8 encoding	37

Contents

Eight bit host	38
Allow lowercase to host	38
Host character set	38
National transliteration (7-bit host)	38
Use SI/SO for printing	38
VT / ANSI – General	39
Destructive backspace	39
Capture on CR (Capture delimiter)	39
Scrollback length	40
Initial screen width	40
Initial screen height (rows)	40
Settings – Screen	41
Toolbar setup	41
Edit	41
Add buttons	42
Add toolbar	42
Options	42
Toolbar options	42
Toolbar 1-n buttons	43
Button size	43
Macro string	43
Screen Options	45
Font size	46
Font name	46
Font style	46
Cursor type	46
3D variable fields	46
Blinking cursor	47
White background 3D fields	47
Show toolbar	47
Limit screen view (x,y)	47
Use double tap to	48
Extra line spacing	48
Force fixed character width	48
Hotspots	48
Screen Colors	49
Color selection	49
Color adjustment	50
ANSI / VT / DKU / 7700 / 7102 attribute mapping	50
Settings – Devices	52
Print options	52

Barcode reader options 53

- Barcode reader device type 53
- More options 54
 - Action after scan 54
 - Remove char(s) at start 55
 - Remove char(s) at end 55
 - Add text before 55
 - Add text after 55
 - Card data format string 55
 - Use keyboard mapping 56
- Application Mode (SPP) for Apple iOS devices 57

Keyboard options 57

- External keyboard 57
- Keyboard mapping 58
- Define macro 59

Distributing Glink configurations from an EMM/MDM..... 61

Download the Glink configuration from a web server 61

Set Glink parameters from the MDM 62

General options..... 66

Passwords 66

Toolbar 67

Remove a session configuration 67

XML Schema 67

MDM Parameter list 69

How to install SSL/TLS client certificates and SSH private key files 87

SSL/TLS Client certificates..... 87

- How to import a PSKC #12 certificate (.p12 file)..... 87
- Distribute the p12 client certificate file to the auto-import folder (Android only) 87
- Use AppConfig parameter in MDM to download p12 client certificate (Android and iOS)..... 87
- Send p12 client certificate as e-mail attachment (Android and iOS) 88

SSH private key file 88

- How to import a private key file 88

How to export/send a debug trace and configuration file 89

Enable, capture and export debug file 89

Export configuration file 91

Contents

The GlinkProxy Session Persistence Server	92
How it works	92
Delivery	93
Install and setup.....	93
Startup	94
Settings	94
Command line parameters	96
Glink for iOS/Android Proxy Server Settings	97

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 VT 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 VT is used to access applications running on Bull, IBM, DEC, UNIX and Linux host systems. Glink VT has emulations for the DEC VT420, VT320, VT340, VT220 and VT102 terminals and ANSI 3.64 terminals.

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

Features

- VT420, VT320, VT340, VT220, VT102, ANSI 3.64 terminal emulation, all screen sizes
- Telnet and SSH 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
- 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
- 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
- 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
- Configurable scrollbar buffer contains history of your host session

Introduction

- Print or e-mail host print data
- Print or e-mail terminal emulation content or scrollbar 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.

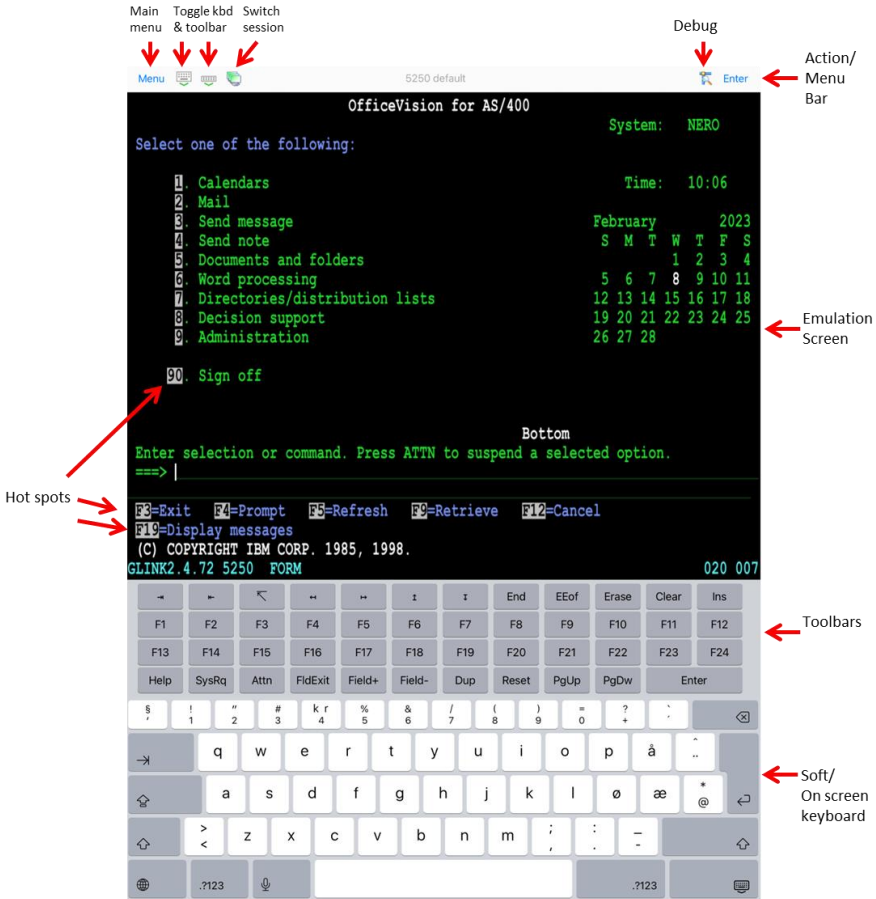
You will also find a description of all available options in this document.

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, Multiline, Single-line and Off. 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 configure 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 address 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 executed.

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 VT emulation:

F11 to F20 : shift+F1 to shift+F10 or alt+F1 to alt+F10

Shortcuts for keyboards without function keys

For 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

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

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.

Getting started

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.

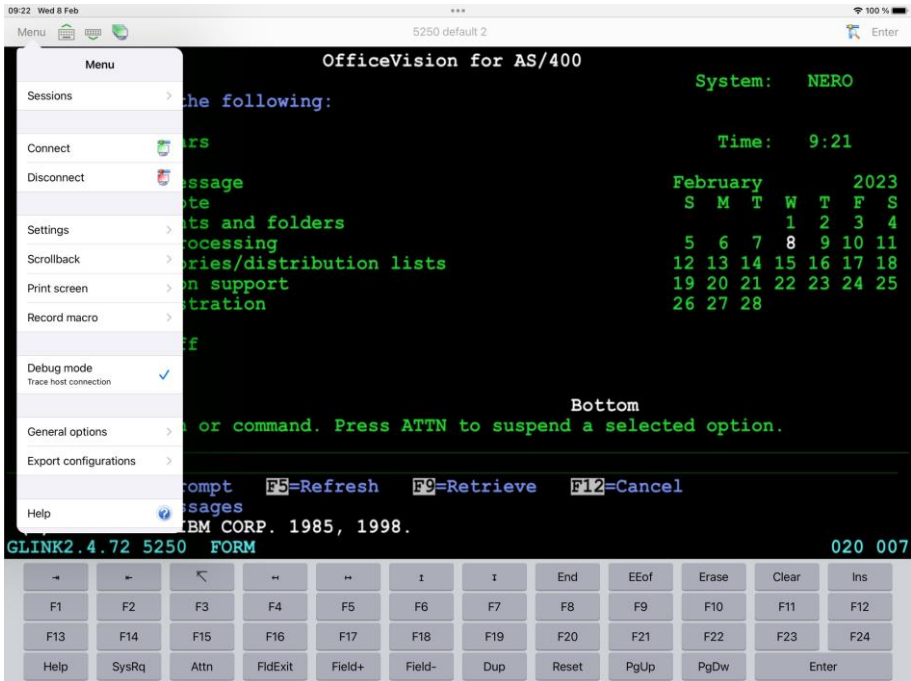
See the section about *The GlinkProxy Session Persistence Server*.

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

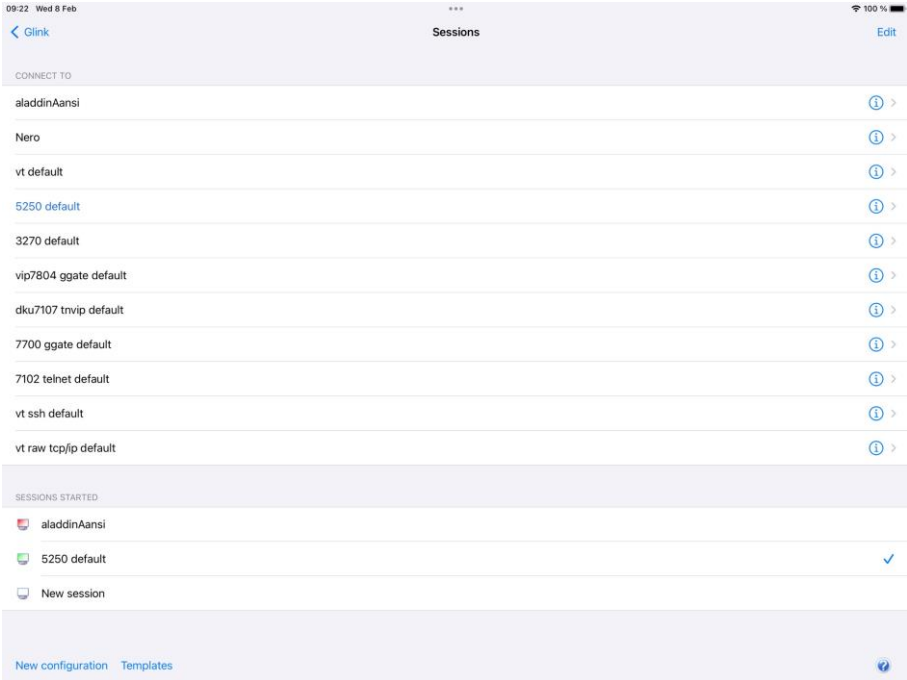
Menu



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 *OK* 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. 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* 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*. The macro will now be saved to the current toolbar. 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. 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.

Menu

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

<i>Off</i>	No click
<i>Sound 1</i>	Click sound 1
<i>Sound 2</i>	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.

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

Telnet options

[← Settings](#) Telnet options 

Server address

Required

Terminal type

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	<input type="checkbox"/>
Server address:	<input type="text"/>
User name:	<input type="text"/>
Password:	<input type="text"/>
Private key file:	<input type="text"/>
Keep alive interval in seconds	<input type="text"/> (NB)

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 must specify a port number, use the following syntax:

[IPv6 address]:port

for example

Settings – Communication

[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.

Settings – Communication

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.

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 attach-

Settings – Communication

ment. 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.

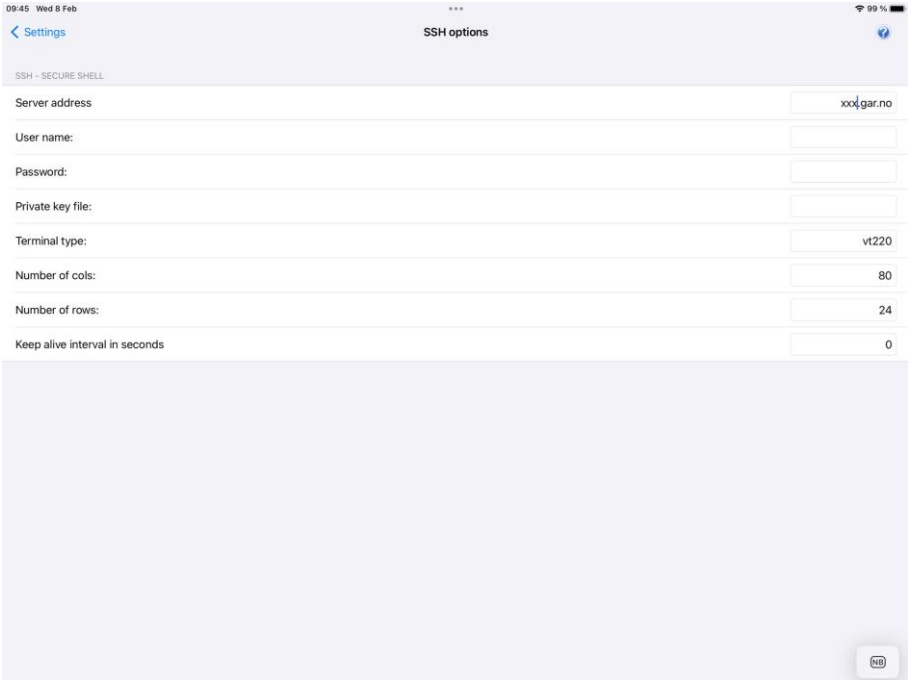
Keep alive interval

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

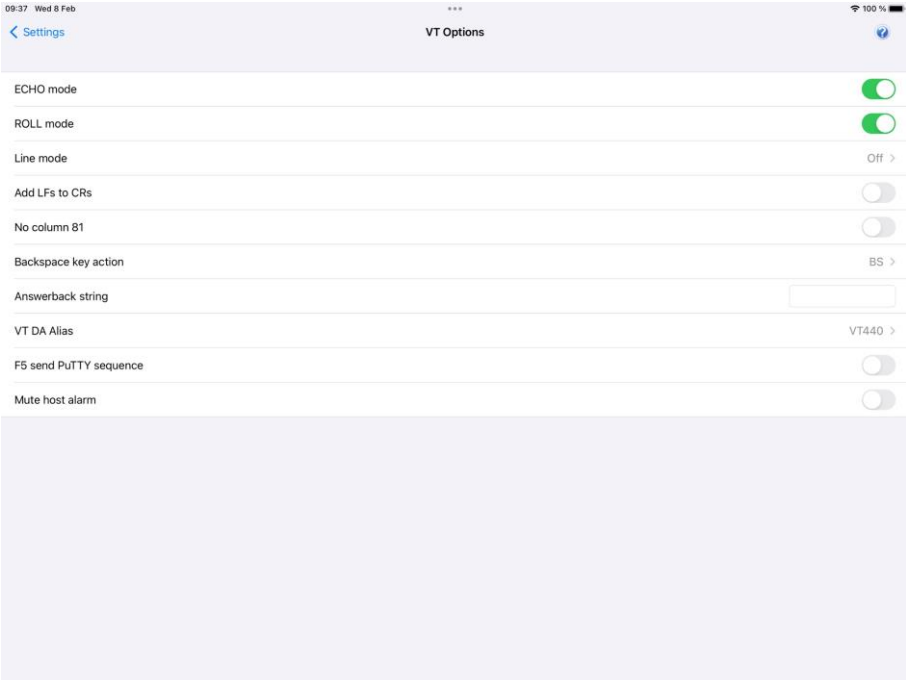
Specify the number of rows for the pseudo terminal, for example 24.

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.

Settings – Emulation

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 in 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

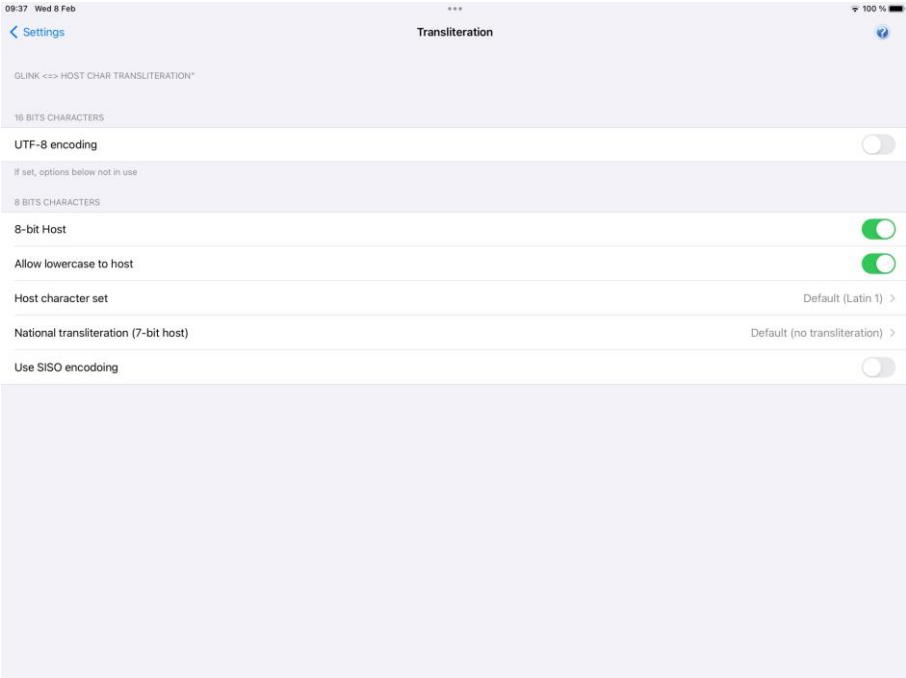
VT DA Alias

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.

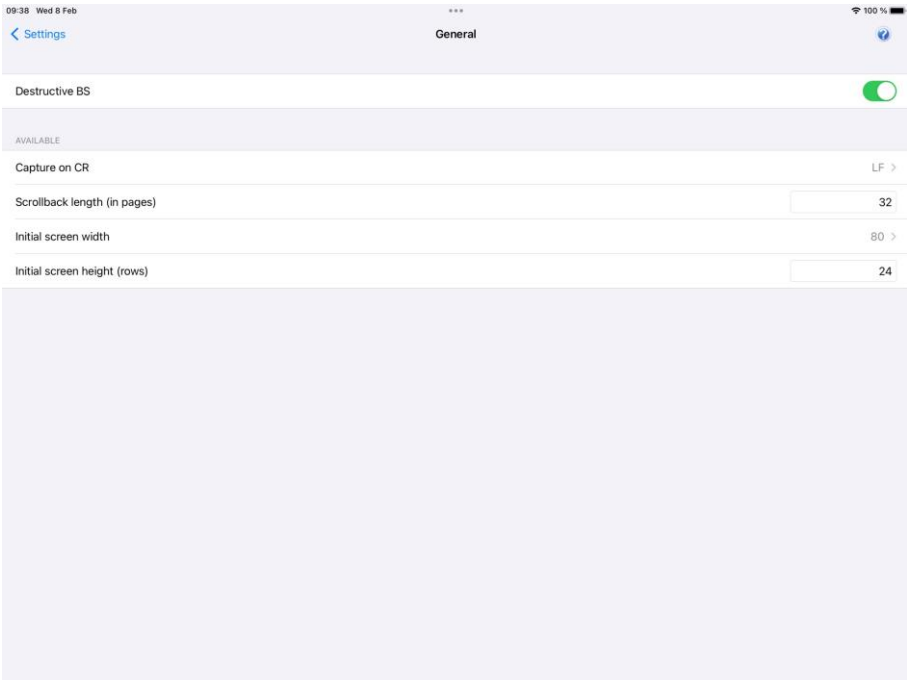
National transliteration (7-bit host)

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



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

Settings – Emulation

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).

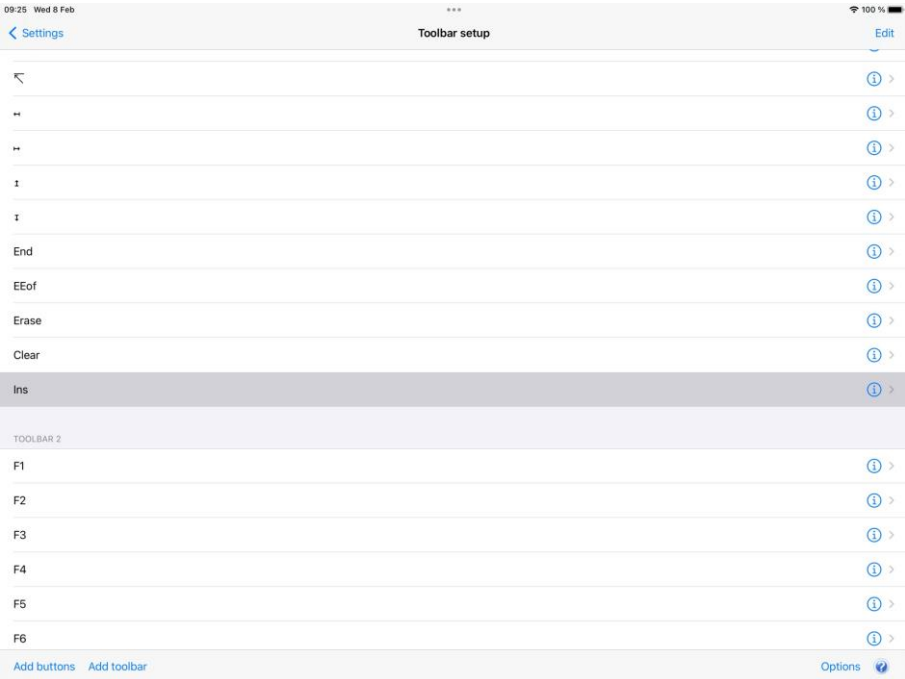
Initial screen height (rows)

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



Edit

The Edit button allow 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 allow 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 below 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 allow 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 adjust 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 *Enter* 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 adjust 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.

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

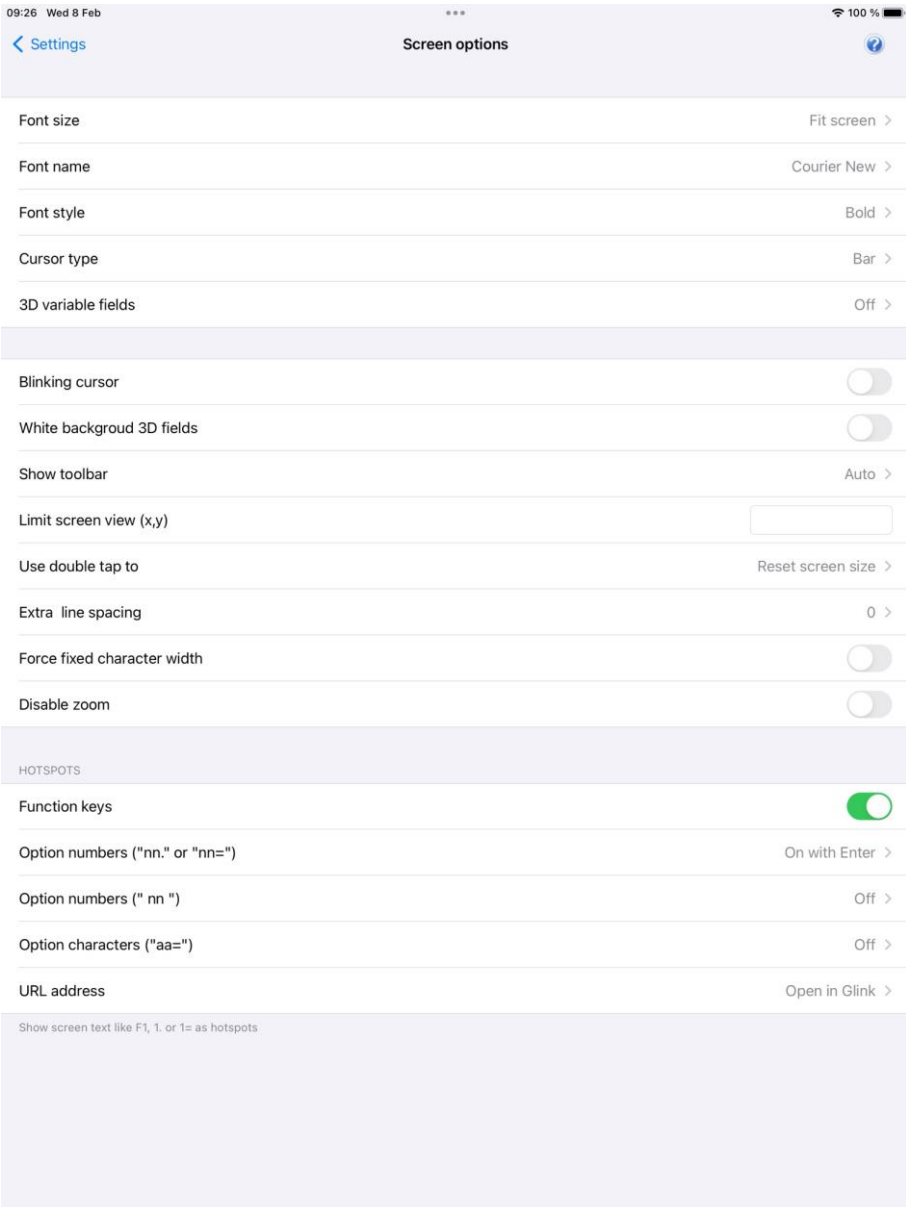
Settings – Screen

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 single-line toolbar with a toggle button to switch between the toolbars defined or as a 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.

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.

Please note that you can change the number of buttons and the number of toolbar lines with the Toolbar setup function.

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.

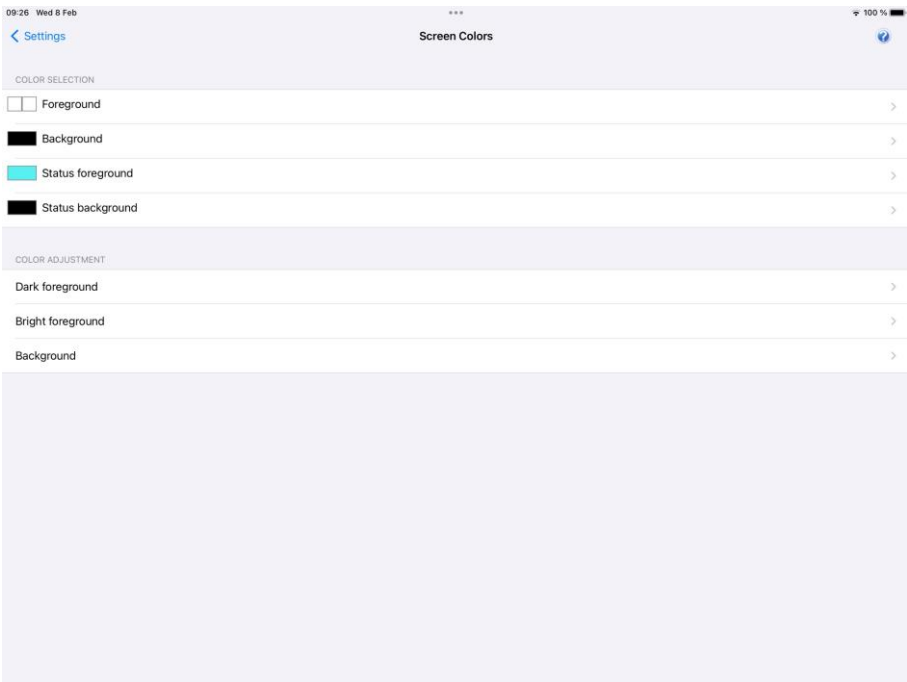
If Option numbers is enabled, text like 2. or 2= will be searched for and highlighted (the number only). If this hotspot is tapped, the number is added at the current cursor position. Optionally it can be followed by the Enter function to transmit it. Numbers larger than 99 is not converted to hotspots.

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

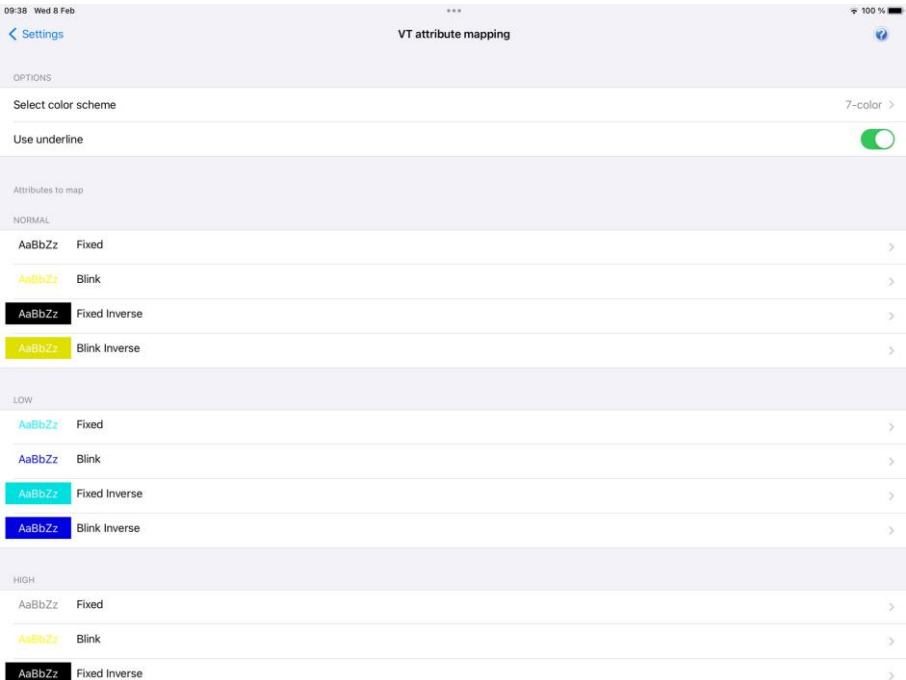
The screen color options allow you to choose the default foreground and background colors for the emulator screen and for the status line.

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.

ANSI / VT / DKU / 7700 / 7102 attribute mapping



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.

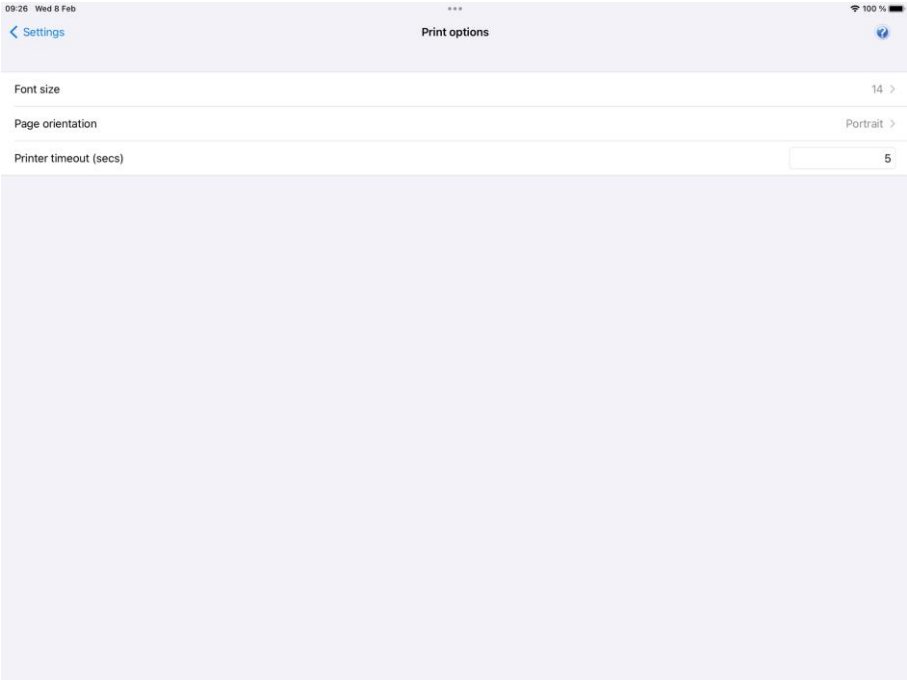
Normally Glink will preserve the state of the underline as it is sent by the host, but if you have mapped underline to a specific color then you may find that to be enough. In that case uncheck the Use underline option.

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

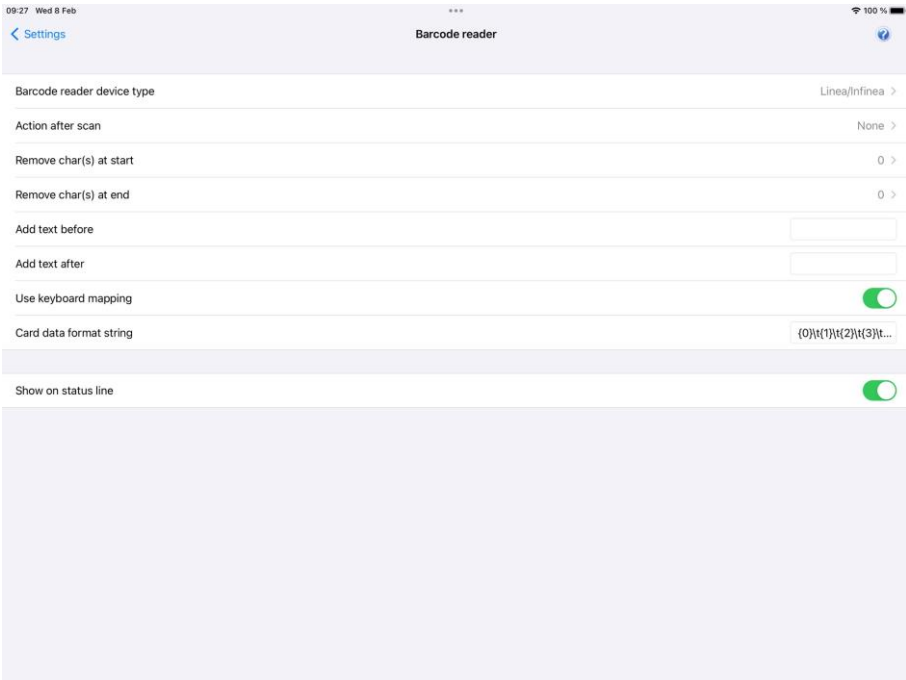
Print options



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 options



Barcode reader device type

Barcode reader device Either HID mode scanner, camera, Socket Mobile Bluetooth scanner or Linea Pro/Infinea Tab (Linea/Infinea may also have a magnetic stripe card reader). The barcode data is inserted as normal input data at the current cursor location of the Glink display.

If the scanner is used as a HID keyboard input device, Glink will not know if the data is coming from the actual keyboard or from the scanner. The options below is not in use for scanners operating in HID mode. Instead, use the configuration options available for the scanner itself.

A bar code icon is added to the toolbar at the top of the Glink display if the Camera option is selected. It is only visible when Glink is connected to the host application. Tap the bar code reader icon to use the camera to scan a barcode.

If the Linea/Infinea option is selected, Glink will accept barcode and magnetic card data as soon as Glink is connected to the host application.

If a Socket Mobile bluetooth barcode Scanner is selected, make sure the scanner is set up to use Application Mode (SPP) for Apple iOS devices.

Settings – Devices

Select the proper barcode reader device type:

Keyboard
Wedge

Default. Scanned data is received as keyboard input. Glink cannot separate scanned data from keyboard input.

Camera

Enables scanning using the camera. An icon is added to the action bar at the top of the Glink display. Tap the icon to use the camera for barcode scanning.

Linea/Infinea

Use with *IPCMobile* (Infinite Peripherals) Linea and Infinea barcode scanner products. Glink will accept both scanned barcodes and Magnetic card data

Socket Mobile

Use with *Socket Mobile Scanner*. Use the Socket Mobile Companion App to pair and set up the scanner to use Application Mode (SPP) for Apple iOS devices.

More options

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

Card data format string

The Card data format string option formats the data read from the Linea/Infinea magnetic stripe card reader. The option is only visible if Linea/Infinea is selected.

The card reader delivers card data to Glink unformatted. Glink splits the data into an array of fields and uses the *Card data format string* to select and format one or more fields. This means that just a part of the magnetic card may be delivered to a host application. An empty *Card data format string* means deliver all the card data unformatted.

{n} is used as syntax to select field n. A format string like
 {0}{1}{2}
 will select the first three fields.

Financial card data will be split into eight fields:

Field	Length	Type
{0}	19	PAN
{1}	26	NAME
{2}	2	YY
{3}	2	MM
{4}	3	Service code
{5}	8	Discretionary data

Settings – Devices

{6}	19	PAN (track 2)
{7}	15	Additional track 2

The format string `{0}{1}{2}` will select the PAN,NAME and YY fields and present them as a combined string.

A TAB character (`\t`) may be added to separate the fields:

```
{0}\t{1}\t{2}
```

The fields may also be separated with spaces or text:

```
{0} expire: 20{2}/{3}
```

will present the PAN field together with the expire date fields:

```
1234567890123456789 expire: 2019/07
```

Use keyboard mapping

Check this option if keyboard mapping should apply 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 text 1234:5678; will be interpreted as key input **1234Tab5678Enter**

Show on status line

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

Application Mode (SPP) for Apple iOS devices

The Socket Mobile device is by default setup in HID-Keyboard mode. In this mode the scanner acts as an external Bluetooth keyboard and scanned data is received in Glink as keyboard input.

A better solution is to download from App Store the **Socket Mobile Companion App** and follow the instructions to pair the scanner with the iOS device and set the scanner to use *Application Mode (SPP) for Apple iOS devices*. Then select Socket Mobile as the Barcode reader device type in Glink.

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.

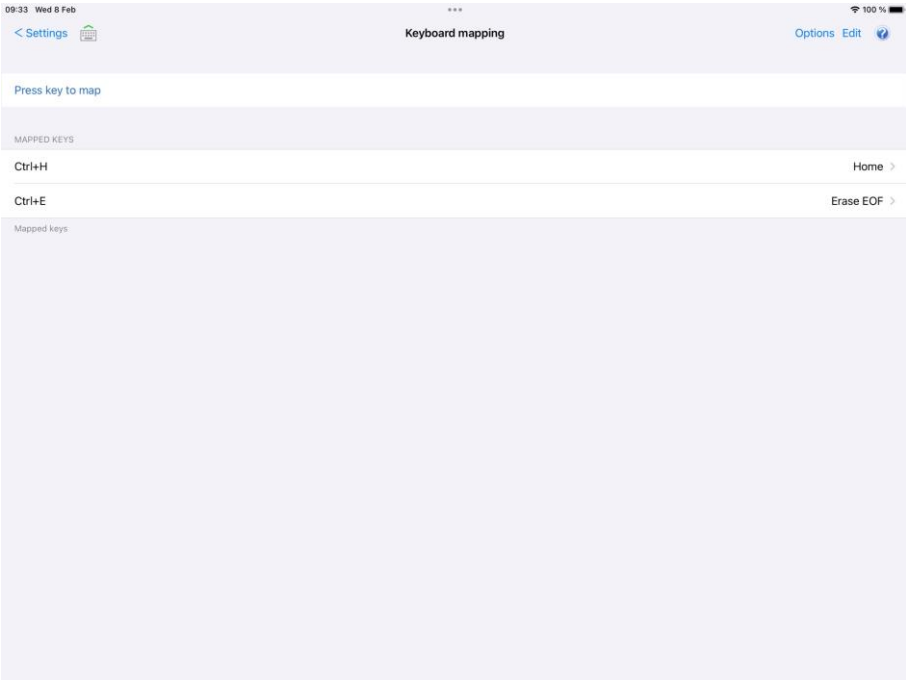
External keyboard

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

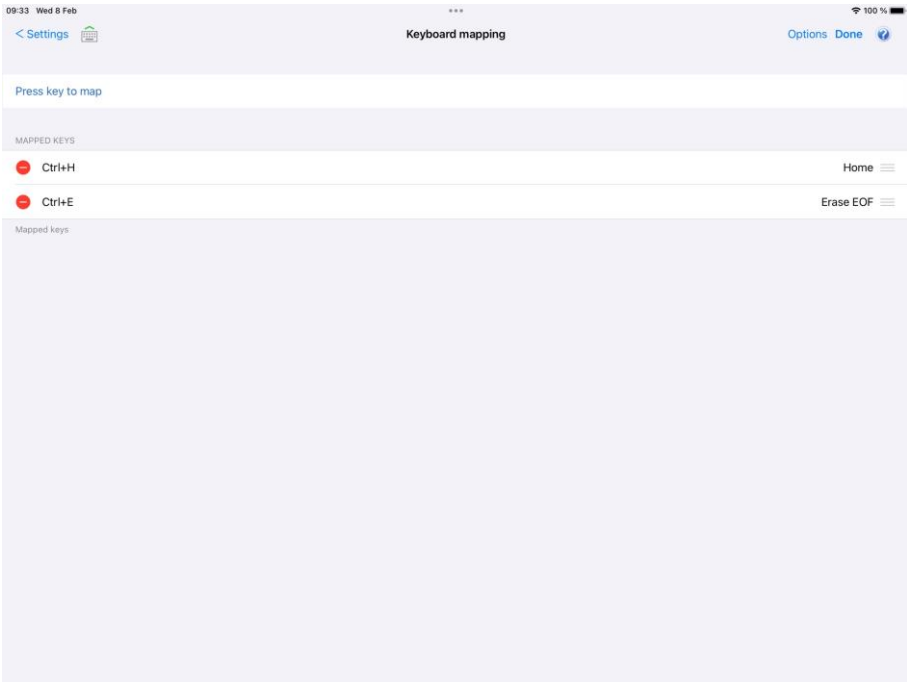


Glink is initially set up with a default keyboard mapping when using an external keyboard. You may override the default mapping or add a new mapping.

The list box displays the keys mapped if any.

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.

Settings – Devices

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.
config.url	This parameter (key-name) provides the URL of the Glink config-

Distributing Glink configurations from an EMM/MDM

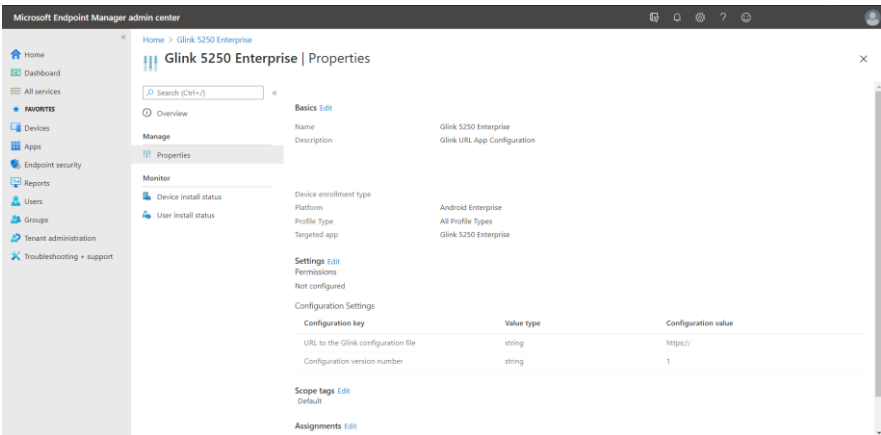
uration file. Please note that this parameter is ignored if the `config.version` parameter is missing or the version number is the same as the version number on the device. So every time you update the configuration file, you must remember to update the `config.version` parameter

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.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.
config.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.servername	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:

Distributing Glink configurations from an EMM/MDM

MobileIron (Android):

The screenshot shows the configuration page for 'Glink 5250 Enterprise' in the MobileIron console. The page includes navigation tabs for 'Details', 'Distribution', 'App Configurations', 'Reviews', and 'App Config Feedback'. The 'App Configurations' section is active, showing 'Managed Configurations' for the application. The configuration details are as follows:

Configuration Name	Value
Configuration version number	2
URL to the Glink configuration file	
Configuration name	AS400
Host address	myhost.mydomain.com

Below the main configuration, there are four 'Additional parameters' sections, each with a 'Parameter key' and a 'Parameter value':

- Parameter key: com.bmcSSUser; Parameter value: myusername
- Parameter key: com.bmcSSOutgoing; Parameter value: #mypassword
- Parameter key: 1_confly_name; Parameter value: AS400-2
- Parameter key: 1_com_servname; Parameter value: myhost2.mydomain.com

MobileIron (iOS):

Distributing Glink configurations from an EMM/MDM

MobileIron CLOUD Dashboard Users Devices Apps Content Configurations Policies Admin

App Catalog Categories Reviews Catalog Settings Distribution Filters Apple Apps and Books

Glink VT Gallagher & Robertson AS | Delegation Status: App is delegated

Details Distribution App Configurations Reviews

App Configurations Summary > iOS Managed App Configuration

Cancel Update

Configuration Setup

Name
My Linux host

+ Add Description

Configuration Source

Source Type
None

iOS Managed App Settings

Key	Value	Type
config.version	1	STRING
config.name	myhost	STRING
com.servername	myhost.mydomain.com	STRING

+ Add | Use .plist

Array type values should be separated by comma (example: 2,33,44) and date value should be in milliseconds (example: 1437496170000).

Distribute this App Config

Choose one of these options

Everyone with App No One Custom

Meraki (Android):

Meraki Search Dashboard

Home > Settings > Managed App Config

Glink 5250 Android

Managed App Config

Platform: Android

App ID: Glink 5250 Enterprise

Screens: Show supported settings

Settings

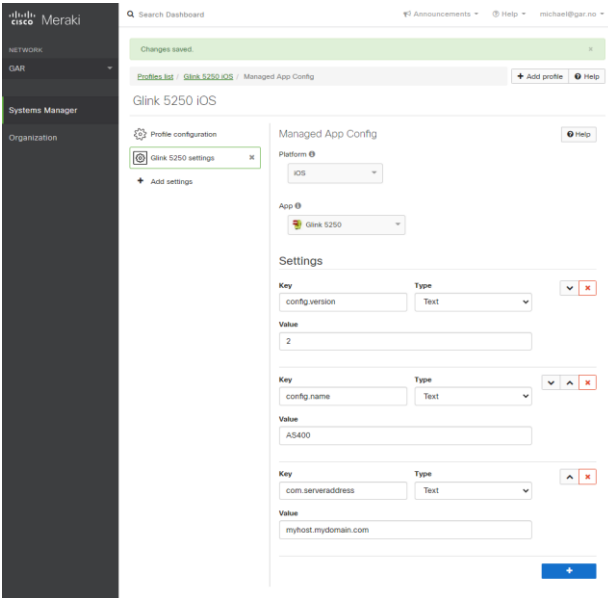
Key	config.version	Configuration server number	8
Type	Text	Non-numeric values require an updated configuration file	
Key	config.name	Configuration name	AS400
Type	Text	Cloudy or mostly a session configuration	
Key	com.servername	Host address	myhost.mydomain.com
Type	Text	Address to host application	
Key	parameters	Additional parameters	parameters.txt

Add parameter

Key	parameter_any	Parameter key	See expected configuration file
Value	com.bank250user	Parameter value	
Key	parameter_value	Parameter key	See expected configuration file
Value	mydomain.com	Parameter value	

Meraki (iOS):

Distributing Glink configurations from an EMM/MDM



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*

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

Distributing Glink configurations from an EMM/MDM

```
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

[The AppConfig Community](#), supported by a collection of industry leading EMM solution providers, 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>
```

Distributing Glink configurations from an EMM/MDM

```
        <value>MyServerAddress</value>
    </defaultValue>
</string>
</dict>
</managedAppConfiguration>
```

Please note that the **bundleId** must match the Glink version you are using; either no.gar.glink5250, no.gar.glink3270, no.gar.glinkvt or no.gar

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	
Remove all toolbar definitions	toolbar.remove	true, false	Boolean
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 millisecc - 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/Receive notification sound - Android only	screen.sndRcvSound	0=Off , 1=On, 2=Send only, 3=Recv only	Integer
Show Sessions view when disconnected -	screen.showSess	true, false	Boolean

Distributing Glink configurations from an EMM/MDM

Android only			
Password	screen.cfgPassword		String
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)			
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

Distributing Glink configurations from an EMM/MDM

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
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

Distributing Glink configurations from an EMM/MDM

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
TN3270 options			
Server address	com.serverName		String
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	com.clientCert		String

Distributing Glink configurations from an EMM/MDM

certificate:			
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.ibm5250IBMMMSGQNAME	QSYSOPR	String
Message library	com.ibm5250IBMMMSGQLIB	*LIBL	String
Font ID	com.ibm5250IBMFONT	11	Integer
Form feed type	com.ibm5250IBMFORMFEE D	A=Auto feed sheets , C=Continuous paper, U=Manual feed sheets	String
TNVIP options			
Server address	com.serverName		String

Distributing Glink configurations from an EMM/MDM

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
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

Distributing Glink configurations from an EMM/MDM

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
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

Distributing Glink configurations from an EMM/MDM

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
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	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	String

Distributing Glink configurations from an EMM/MDM

		Kingdom, US United States	
Use SISO encoding	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
Scrollbar 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

Distributing Glink configurations from an EMM/MDM

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
Scrollbar 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

Distributing Glink configurations from an EMM/MDM

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

Distributing Glink configurations from an EMM/MDM

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

Distributing Glink configurations from an EMM/MDM

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, SourceCodePro-Regular, SourceCodePro-Semi Bold, SourceCodePro-Bold iOS: CourierNew , 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	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)	Integer

Distributing Glink configurations from an EMM/MDM

		Android only, 5==Enable/Disable touchscreen (Disabled) Android only	
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
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 foreground			
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	String

Distributing Glink configurations from an EMM/MDM

		config	
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 foreground			
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

Distributing Glink configurations from an EMM/MDM

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, 10=Cipherlab, 9=Datalogic, 4=Denso Scanner, 5=Honeywell Scanner, 11=M3 Mobile, 12=Point Mobile, 6=Socket Mobile, 3=Unitech Scanner, 13=Urovo scanner, 2=Zebra skanner iOS: 0=Keyboard Wedge, 1=Camera, 2=Linea/infinia, 3=Socket Mobile	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
Add text after	print.bcAddLast		String

Distributing Glink configurations from an EMM/MDM

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
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 <u>config</u> 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 (Android and 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...
```

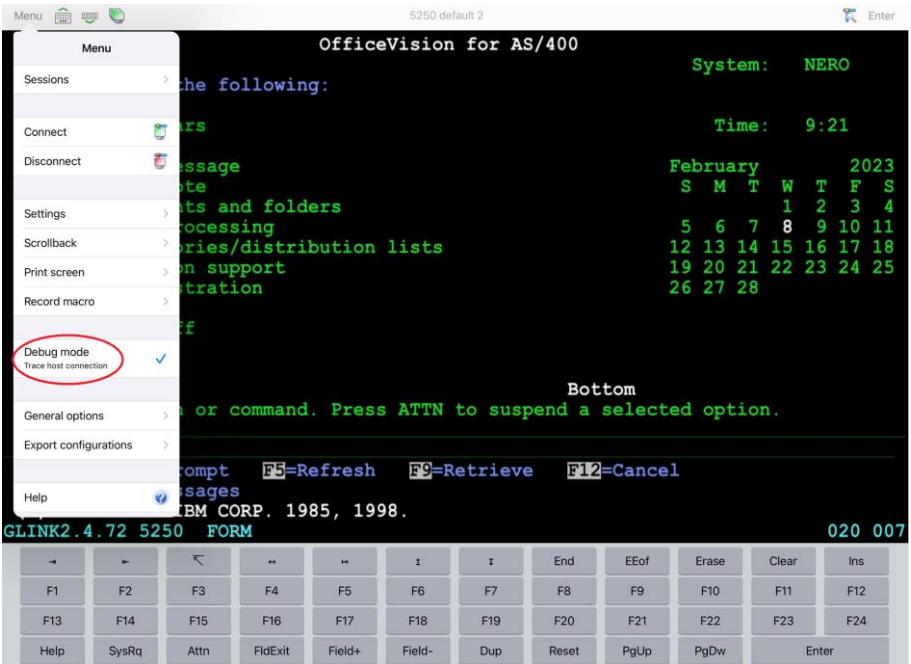
Glink will create a file with the file name *my_pk_file* with the content supplied.

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**

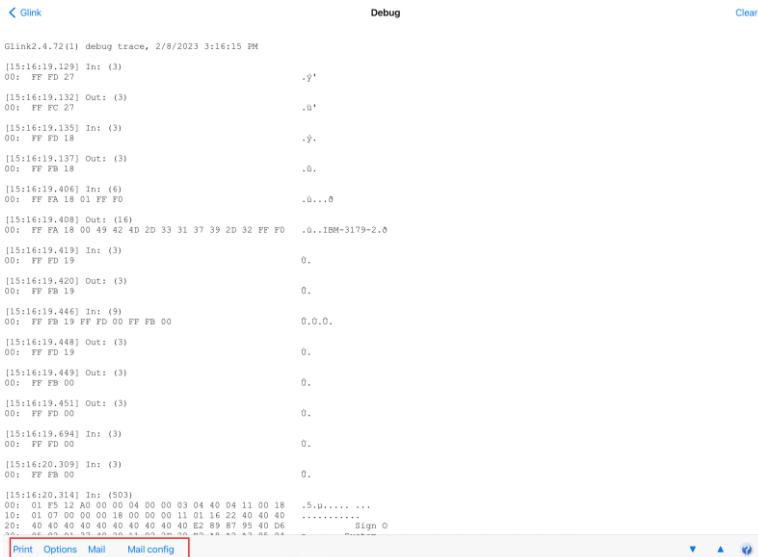


2. Return to emulation screen and tap **Disconnect** + **Connect**
3. Interact with the host application until you have collected the request information

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.



How to export/send a debug trace and configuration file

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 attachment: **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

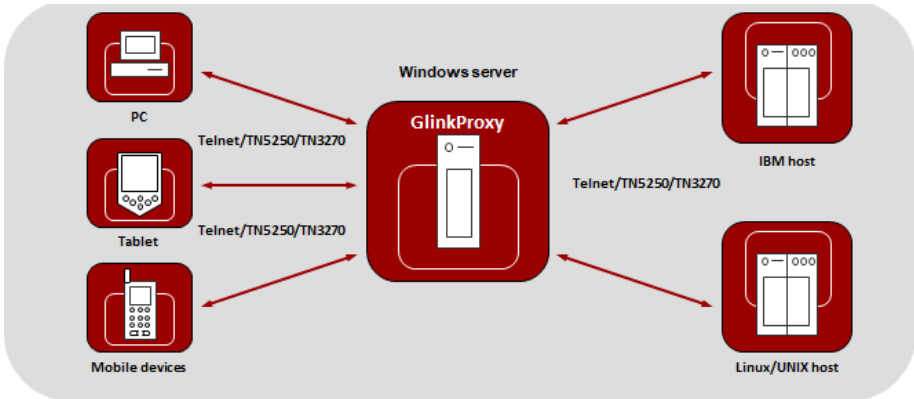
GlinkProxy runs on a Windows server, can be located close to the host computer, and is designed to extend the life of sessions between Glink terminal emulation clients and hosts. A device running Glink might go idle, enter power-saving mode or move out of WiFi range and prematurely terminate the session 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

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

The GlinkProxy Session Persistence Server

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.

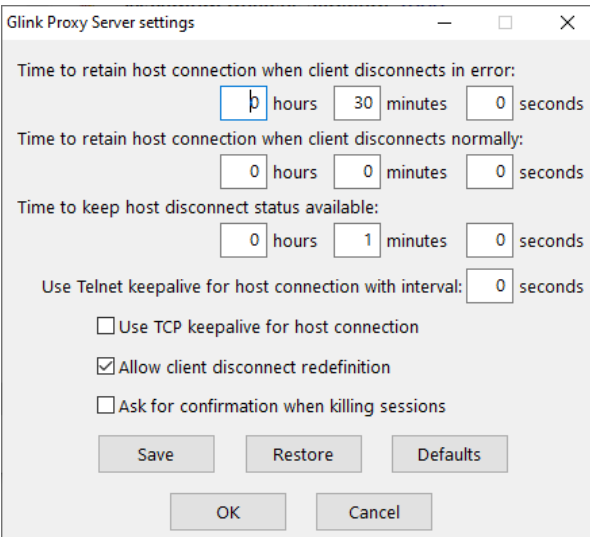
Below you can read more about the Settings menu, its parameters and how you can set them from the command prompt. You will also find a description of the proxy server parameters that can be set in Glink for iOS and Glink for Android.

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.



The screenshot shows a dialog box titled "Glink Proxy Server settings". It contains several configuration options:

- Time to retain host connection when client disconnects in error:** 0 hours, 30 minutes, 0 seconds.
- Time to retain host connection when client disconnects normally:** 0 hours, 0 minutes, 0 seconds.
- Time to keep host disconnect status available:** 0 hours, 1 minutes, 0 seconds.
- Use Telnet keepalive for host connection with interval:** 0 seconds.
- Use TCP keepalive for host connection
- Allow client disconnect redefinition
- Ask for confirmation when killing sessions

At the bottom, there are buttons for "Save", "Restore", "Defaults", "OK", and "Cancel".

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.

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

The GlinkProxy Session Persistence Server

		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 debug	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:

The GlinkProxy Session Persistence Server

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
Connection 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.