

---

# Wireless Markup Language (WML)

Wireless Markup Language (WML) offers a text-based method of designing a menu structure for the display screen of selected printers. By leveraging Set/Get/Do (SGD) and files containing Zebra Programming Language (ZPL) commands, customized menus can be created.

## Contents

WML Overview . . . . .	1340
WML Details . . . . .	1340
Supported Printers . . . . .	1341
Professional Services for WML Content Creation . . . . .	1342
WML Tags . . . . .	1342
Using WML . . . . .	1342
WML Examples . . . . .	1347
Troubleshooting Scenarios . . . . .	1355

## WML Overview

Wireless Markup Language (WML) offers a text-based method of designing customized menus on the LCD front panel of selected printers. By leveraging Set-Get-Do (SGD) and Zebra Programming Language (ZPL) commands, menus that feature both display and command features can be created. The WML “card” structure makes it possible to link from one menu screen to another, creating menus that are as many levels “deep” as desired or reduced to only those options needed by the printer operator.

For details on SGD commands, see [SGD Printer Commands on page 631](#). For details on ZPL commands, see [ZPL Commands on page 47](#).

## WML Details

A WML file is made up of tags, which are similar to HTML tags. For a list of the supported WML tags, see [WML Tags on page 1342](#).

Using WML on the printer is dependent on the presence of a single `index.wml` file, stored in the printer's E: memory. The `index.wml` file can contain one or more “cards”, with each card defining the content of a single menu. Everything within the card tag (`<card> </card>`) constitutes one complete front panel menu. Cards can also contain hyperlinks to other menus. If the `index.wml` has three cards, with links between the cards, that means there are three front panel menus. It is also possible to create multiple `.wml` files, with links between them and the `index.wml` file. In cases where multiple `.wml` files are used, it is recommended that each file should be structured to provide a link back to the main menu as described in the `index.wml` card.



**Note** • Only one `index.wml` file can reside on a printer at any time

WML defined menus can use Set-Get-Do (SGD) commands to retrieve or set printer settings. For example, a menu might display the printer’s current baud rate, while also offering other potential baud rate settings for the printer selection. In more advanced uses, WML defined menus can cause ZPL command files, stored in the printer E: memory, to be injected into the printers command engine – where they will be read in and acted upon. In this use, the ZPL command file files are known as `.nrd` files.

For example, a WML defined menu could call an `.nrd` file that contains a customized set of printer configuration commands. In this way, different profiles can be created for the printer - making it possible for the printer operator to select the appropriate configuration profile needed for the task the printer is being used in.

An important concept to consider is that the WML menu completely defines what is displayed on the printers screen. If an item is not included in the WML menu definition it will not be displayed to the user.



**Note** • The `index.wml` file must reside on the printer's E: drive for the WML menu to display. If the `index.wml` file is on a drive other than E:, then the standard front panel menus display.



**Note** • When a WML menu is resident on the printer, the standard menu system can be easily be accessed by holding down the Cancel and Setup/Exit buttons (on the ZM400) or the Cancel and Setup/Exit buttons (on Xi4) or the Select button (on GX) on the front panel while the printer powers up. Hold the buttons down until the PRINT READY message displays on the front panel. To return to the WML defined menu, reset the printer again.

## Supported Printers

WML is supported on the following printers, using the indicated firmware. The buttons on the printers' front panel that are used for Navigating WML defined menus are noted.



**Note** • When a WML defined menu is in use, the stripes pattern found at the top of selected printers is not displayed.

**Table 57 • WML-Supported Printers**

Printer	Firmware	Number of "lines" available	Menu Navigation Buttons	Keys to Access Standard Menu System
105SLPlus	V53.17.15 Z (or later)		•	
Xi4™ series	V53.17.5Z (or later)	5	Select + (PLUS) - (MINUS)	Hold down CANCEL & SETUP/EXIT during power-up
ZM400™	V53.17.5Z (or later)	5	Select + (PLUS) - (MINUS)	Hold down CANCEL & SETUP/EXIT during power-up
ZM600™	V53.17.5Z (or later)	5	NEXT/SAVE + (PLUS) - (MINUS)	Hold down CANCEL & SETUP/EXIT during power-up
G™ -series	V56.17.5Z (or later)	4	SELECT SCROLL	Hold down SELECT during power-up

## Professional Services for WML Content Creation

Zebra offers a Professional Services group that can help with the creation of WML content. To inquire about Zebra’s Professional Services, please contact your Zebra account representative.

### WML Tags

Table 1 shows the WML tags and tag parameters that can be used to create a menu system. As with other tag-based languages, such as HTML and XML, ending tags should be used to indicate the end of a structure. An example of an ending tag would be `</wml>`, which indicates the end of a WML script.

### Using WML



**Important** • Using end tags is required to create well formed and functional WML scripts.

**Table 58 • WML Tag Descriptions**

<code>&lt;wml&gt; &lt;/wml&gt;</code>	indicates the beginning/end of the WML script
<code>&lt;display&gt; &lt;/display&gt;</code>	indicates the beginning/end of the content to display on-screen
<code>&lt;card&gt; &lt;/card&gt;</code>	indicates the beginning/end of a card
<code>&lt;p&gt; &lt;/p&gt;</code>	indicates the beginning/end of a Paragraph
<code>&lt;br/&gt;</code>	Line break
<code>&lt;a href="#menu"&gt;Menu&lt;/a&gt;</code>	Hyperlink to another card
<code>&lt;timer value="xx"&gt; &lt;/timer&gt;</code>	Controls display timer in 10 <sup>th</sup> of a second increments
<code>" ontimer="#main"</code>	Controls action to take at timer end
<code>alerts="on"</code>	Controls display of on-screen alerts
<code>\$(command.command)</code>	\$ executes a SGD “get” command
<code>&lt;do .....&gt;&lt;setvar .....&gt;&lt;/do&gt;</code>	Controls execution of do and setvar commands

This section provides you with the necessary steps to prepare and transmit WML content to the printer. There are two methods to send WML content to the printer - via the FTP protocol or using the “CISDFCRC16” command. Both methods are detailed below.



**Important** • The & (ampersand) character should not be used within the body of any Paragraph tag (`<p>`). If an ampersand is present within the body of a Paragraph tag, a WML-based menu may not function as expected.

## Create a Sample index.wml File:

1. Open a text editor.
2. Type (or copy/paste) the following text:

```
<wml>
<display>
  <card>
    <p>Hello World!!</p>
  </card>
</display>
</wml>
```

3. Save this file with this name: `index.wml`.

## Prepare the Printer to Receive WML Content via FTP:

WML files – and any .nrd files used by a WML menu structure – must be stored in the printers E: memory location. While the files are first being transmitted to the printer, they should not be processed by the printers ZPL formatting engine. This can be done by configuring the SGD settings "ip.ftp.enable" and "ip.ftp.execute\_file".

The "ip.ftp.enable" setting allows the printer to receive content via the FTP protocol. The "ip.ftp.execute\_file" setting controls the printers' ability to process or not process commands received via the FTP protocol using the printers ZPL engine. By default, both settings are enabled.

1. Set "ip.ftp.enable" to "on" and the "ip.ftp.execute\_file" to "off".

### To do this, send these commands to the printer:

```
! U1 setvar "ip.ftp.enable" "on"
! U1 setvar "ip.ftp.execute_file" "off"
```

2. To confirm these commands are correctly set, send the getvar command to check the settings. To do this, send these commands to the printer:

```
! U1 getvar "ip.ftp.enable"
! U1 getvar "ip.ftp.execute_file"
```

If a terminal emulation program is being used, the following response should be returned from the printer.

```
"on" "off"
```

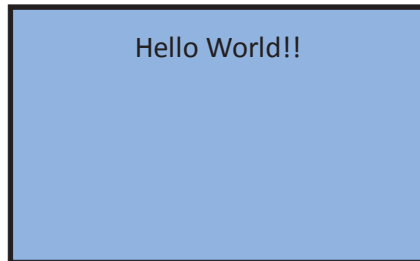


**Note** • Only printers using the Internal 10/100 wired or Internal Wireless Plus & Wireless Plus print server can use the `! U1 setvar "ip.ftp.execute_file" "off"` command. For other print servers, use the "CISDFCRC16" command method detailed below.

## Send WML Content to the Printer via FTP

1. Go to a command prompt.
2. At the command line prompt, type "`ftp xxx.xxx.xxx.xxx`", where `xxx.xxx.xxx.xxx` is the IP Address of the printer. For example, if the IP Address of the printer is 10.3.5.34, the command would be:  
`ftp 10.3.5.34`
3. Press Enter to connect to the printer.
4. Press Enter to log in to the printer.
5. At the FTP prompt, type "`put index.wml`" and press Enter. The `index.wml` file will be transferred to the printer's E: memory.
6. Type "`quit`" to disconnect from the printer and exit FTP.
7. Power cycle the printer.

Once the printer completes the power cycle the display should look similar to this:



For additional `index.wml` examples, see [WML Examples on page 1347](#).



**Note** • When a WML menu is resident on the printer, the standard menu system can be easily be accessed by holding down the Cancel and Setup/Exit buttons (on the ZM400) or the Cancel and Setup/Exit buttons (on Xi4) or the Select button (on GX) on the front panel while the printer powers up. Hold the buttons down until the PRINT READY message displays on the front panel. To return to the WML defined menu, reset the printer again.



**Important** • When using the "`ip.ftp.execute_file`" command, be sure to reset the command back to "on" for use in production processes. If the setting is left in the "off" configuration, when label formats or firmware are sent to the printer via FTP they will not be processed as intended – and the E: memory location can quickly become full.

## Resetting the "ip.ftp.execute\_file" setting

1. To reset the "ip.ftp.execute\_file" setting to the default state, send the following command to the printer.

```
! U1 setvar "ip.ftp.execute_file" "on"
```

## Sending WML Content to the Printer via the CISDFCRC16 Command:

WML files – and any .nrd files used by a WML menu structure – must be stored in the printers E: memory location. While the files are first being transmitted to the printer, they should not be processed by the printers ZPL formatting engine. This can be done by using the CISDFCRC16 command. This command allows content to be written directly to the E: memory location, without being processed by the printers ZPL formatting engine. By using the CISDFCRC16 command, WML content can be transmitted to the printer via the Serial, USB or Parallel ports.

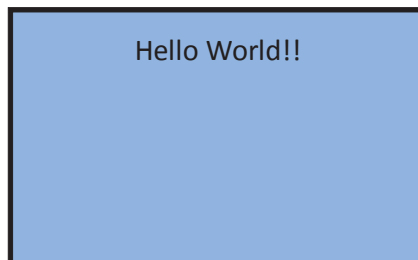
For additional information on the CISDFCRC16 command, see [page XX](#).

1. To send the sample index.wml shown earlier, send the following commands to the printer:

```
! CISDFCRC16
0000
INDEX.WML
0000004E
0000
<wml>
<display>
  <card>
    <p>Hello World!!</p>
  </card>
</display>
</wml>
```

2. Power cycle the printer.

Once the printer completes the power cycle the display should look similar to this:



For additional index.wml examples, see [WML Examples on page 1347](#).



**Note** • When a WML menu is resident on the printer, the standard menu system can be easily be accessed by holding down the Cancel and Setup/Exit buttons (on the ZM400) or Cancel and Setup/Exit buttons (on Xi4) on the front panel while the printer powers up. Hold the buttons down until the PRINT READY message displays on the front panel. To return to the WML defined menu, reset the printer again.

## Retrieving WML Content from the Printer using the file.type Command:

It is possible to retrieve .wml file content from the printer using the "file.type" SGD command. To do this, open a terminal emulation connection to the printer and issue the command. For example, to retrieve the contents of the INDEX.WML file, use the following command:

```
! U1 setvar "file.type" "E:INDEX.WML"
```



**Note** • The file.type command is case sensitive – if the file is stored on the printer as INDEX.WML, the command must use that same case. Additionally, please note that .nrd files are treated as confidential – they cannot be retrieved from the printer.

## Using .nrd Files from WML Menus

It is possible to have a WML menu send the contents of a ZPL or SGD file to the printer to be processed. In this way, the WML menu can leverage complex command scripts in response to the user pressing a single button on the printer. One possible use for this capability would be to create a series of “profile” files that contain all the settings necessary to reconfigure the printer for different uses.

The command files are known as “.nrd” files and are stored directly on the printers E: memory location. The .nrd files can be created using a standard text editor and sent to the printer via FTP or using the CISDFCRC16 command. Files should have an “.nrd” extension.

## Removing WML or .nrd Files from the Printer using the file.delete Command:

It is possible to remove .wml files from the printer using the file.delete SGD command. To do this, open a terminal emulation connection to the printer and issue the command. For example, to remove the INDEX.WML file, use the following command:

```
! U1 do "file.delete" "E:INDEX.WML"
```



## WML Examples

The examples shown below “build” from a simple, display-only, WML menu to a more complex interactive example that uses .nrd files containing ZPL commands. In the initial examples, all lines are explained in detail, in the later examples only the new concepts are covered in detail.

Indenting is used in the examples below to improve readability, it is not necessary in actual use.

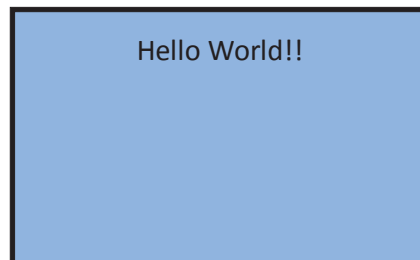
### Example 1

This example shows a basic WML menu structure that uses only fixed text. The content below shows the WML script plus numbered callouts and a table that identify the function of each of the WML tags.

```
1→ <wml>
2→ <display>
3→   <card>
4→     <p>Hello World!!</p>
5→   </card>
6→ </display>
7→ </wml>
```

1	Beginning of the WML file.
2	Beginning of the content to be displayed.
3	The <code>&lt;card&gt;</code> tag begins the definition of this menu.
4	The <code>&lt;p&gt;</code> beings a paragraph, here displaying Hello World! The <code>&lt;/p&gt;</code> ends the paragraph
5	The <code>&lt;/card&gt;</code> tag ends the definition of this menu.
6	End of the content to be displayed.
7	End of the WML file.

In use, this WML menu looks similar to this:



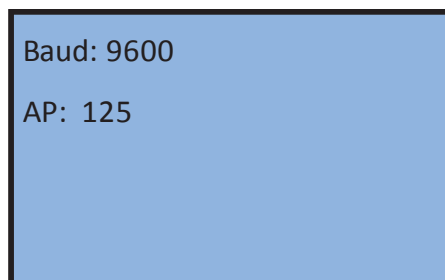
## Example 2

This example demonstrates a WML menu structure that uses fixed text, plus two SGD commands to display the current printer settings for the Baud rate and ESSID settings on the printer.

```
1→ <wml>  
2→ <display>  
3→   <card>  
4→     <p>Baud: $(comm.baud)</p>  
5→     <br/>  
6→     <p>AP: $(wlan.essid)</p>  
7→   </card>  
8→ </display>  
9→ </wml>
```

1	Beginning of the WML file.
2	Beginning of the content to be displayed.
3	The <card> tag begins the definition of this menu.
4	The <p> begins a paragraph. 'Baud:' displays the text Baud: \$(comm.baud) retrieves and displays the printers' current baud rate. The </p> ends the paragraph.
5	A line break
6	The <p> begins a paragraph. "AP:" displays the text AP: \$(wlan.essid) retrieves and displays the printers' current ESSID setting. The </p> ends the paragraph.
7	The </card> tag ends the definition of this menu.
8	End of the content to be displayed.
9	End of the WML file

In use, this WML menu looks similar to this:



## Example 3

This example demonstrates a WML menu structure with two menus. Fixed text and SGD commands are used to display the current printer settings for the Baud rate and ESSID settings on menu one and the Firmware version and ZBI State on menu two. Through use of the ‘timer’ setting, the menu will automatically return to a defined WML card if no buttons are pressed after a set time period. The menu is configured to allow printer alerts (such as HEAD OPEN) to be displayed.

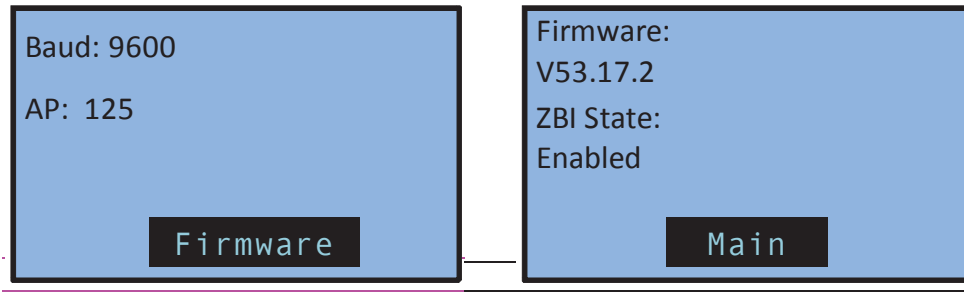
```

1→ <wml>
2→ <display>
3→   <card id="main" title="" ontimer="#main" alerts="on">
4→     <timer value="50"></timer>
5→     <p>Baud: $(comm.baud)</p>
6→     <br/>
7→     <p>AP: $(wlan.essid)</p>
8→     <p> </p><br/>
9→     <p> </p><br/>
10→    <p>   <a href="#system">Firmware</a></p>
11→    </card>
12→    <card id="system" title="" ontimer="#main" alerts="on">
13→      <timer value="50"></timer>
14→      <p>Firmware:</p><br/>
15→      <p>$(appl.name)</p><br/>
16→      <p>ZBI State:</p><br/>
17→      <p>$(zbi.key)</p><br/>
18→      <p>   <a href="#main">Main</a></p>
19→    </card>
20→ </display>
21→ </wml>

```

3	<code>&lt;card id="main"</code> - defines the card's id - "main". <code>title=""</code> - defines the title (not displayed on screen). <code>ontimer="#main"</code> - defines the WML card to display when the timer runs out. <code>alerts="on"&gt;</code> - enables the alerts display feature.
4	<code>&lt;timer value="50"&gt;&lt;/timer&gt;</code> - sets the timer to 50 (in 10 <sup>th</sup> of a second increments).
10	<code>&lt;p&gt;   &lt;a href="#system"&gt;Firmware&lt;/a&gt;&lt;/p&gt;</code> - defines a link to the “system” card.
12	<code>&lt;card id="system"</code> - defines the card's id - "system". <code>title=""</code> - defines the title (not displayed on screen). <code>ontimer="#main"</code> - defines the WML card to display when the timer runs out. <code>alerts="on"&gt;</code> - enables the alerts display feature.
13	<code>&lt;timer value="50"&gt;&lt;/timer&gt;</code> - sets the timer to 50 (in 10 <sup>th</sup> of a second increments).
18	<code>&lt;p&gt;   &lt;a href="#main"&gt;Main&lt;/a&gt;&lt;/p&gt;</code> - defines a link to the “main” card.

In use, these two WML menus look similar to this:



**Note** • GX series printers can display four lines of text. If you are using a GX series printer, remove one line of text from each “card” to use this example.

## Example 4

This example demonstrates a WML menu structure that creates two menu screens and a link to a command file – “config.nrd” – that contains a ZPL command that will cause the unit to print a configuration label.

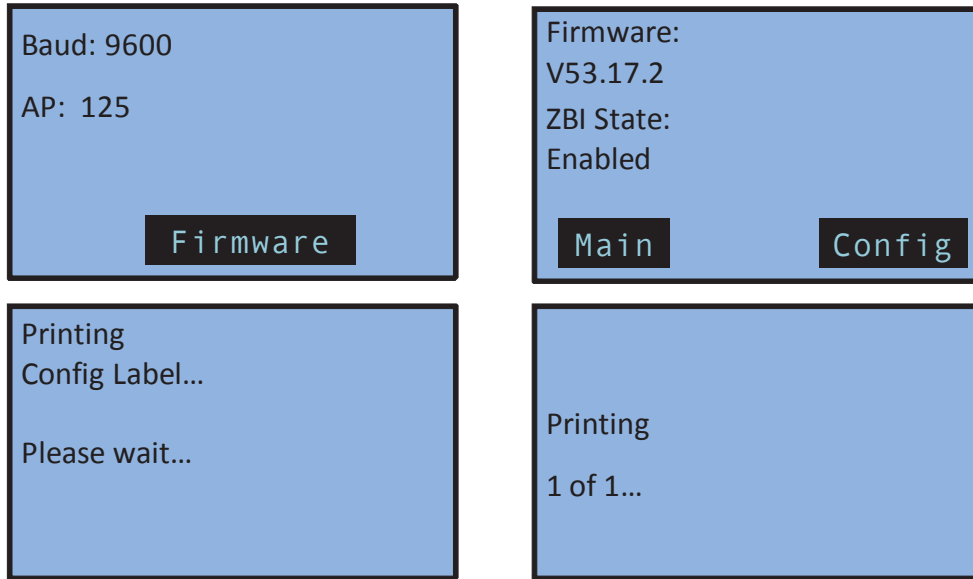
```

1→ <wml>
2→ <display>
3→   <card id="main" title="" ontimer="#main" alerts="on">
4→     <timer value="50"></timer>
5→     <p>Baud: $(comm.baud)</p>
6→     <br/>
7→     <p>AP: $(wlan.essid)</p>
8→     <p> </p><br/>
9→     <p> </p><br/>
10→    <p>   <a href="#system">Firmware</a></p>
11→    </card>
12→    <card id="system" title="" ontimer="#main" alerts="on">
13→      <timer value="50"></timer>
14→      <p>Firmware:</p><br/>
15→      <p>$(appl.name)</p><br/>
16→      <p>ZBI State:</p><br/>
17→      <p>$(zbi.key)</p><br/>
18→      <p><a href="#main">Main</a>
19→        <a href="#config">Config</a></p>
20→    </card>
21→    <card id="config" title="" ontimer="#main" alerts="on">
22→      <timer value="50"></timer>
23→      <p>Printing </p><br/>
24→      <p> Config Label...</p><br/>
25→      <p></p><br/>
26→      <p>Please wait...</p><br/>
27→      <setvar name="file.run" value="e:config.nrd"/>
28→    </card>
29→  </display>
30→ </wml>

```

18	<pre>&lt;p&gt;&lt;a href="#main"&gt;Main&lt;/a&gt;   &lt;a href="#config"&gt;Config&lt;/a&gt;&lt;/p&gt;</pre> <ul style="list-style-type: none"> <li>Defines two links, positioned next to each other - to the "main" and "config" WML cards</li> </ul>
26	<pre>&lt;setvar name="file.run" value="e:config.nrd"/&gt;</pre> <ul style="list-style-type: none"> <li>Defines that the SGD command "file.run" should be used on the "e:config.nrd" file.</li> <li>In this instance, the "e:config.nrd" file contains a single ZPL command - "~wc"</li> </ul>

In use, these WML menus look similar to this:



**Note** • GX series printers can display four lines of text. If you are using a GX series printer, remove one line of text from each “card” to use this example.

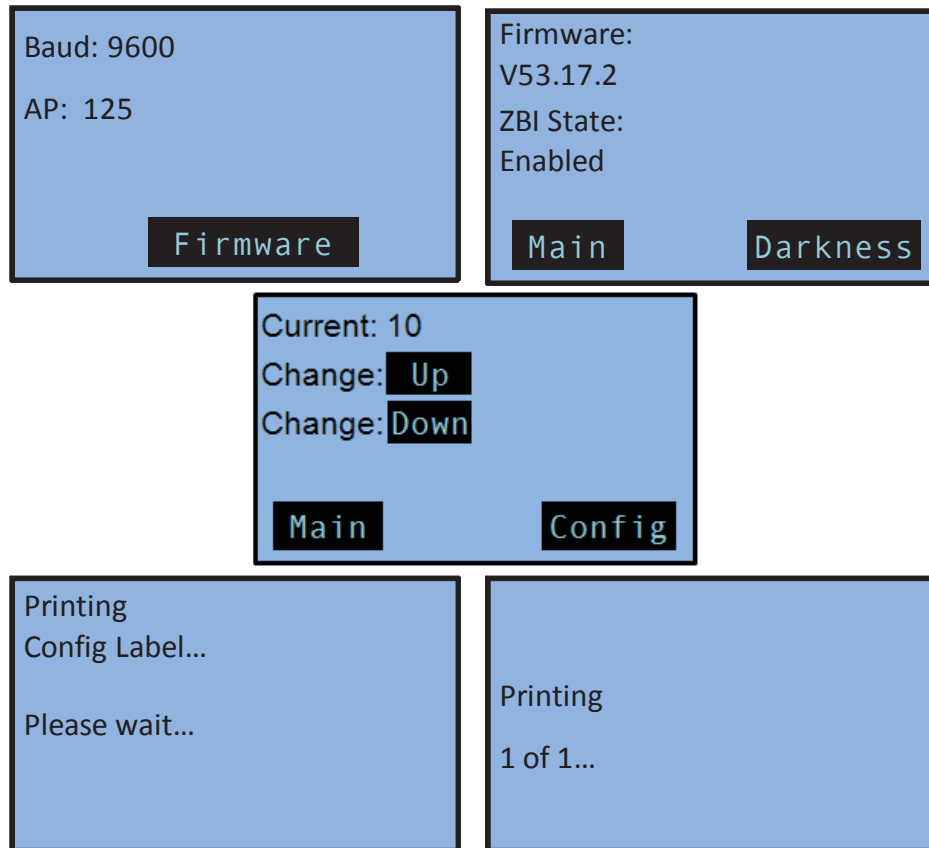
## Example 5

This example demonstrates a WML menu structure with three cards. The "darkness" card leverages WML and the SGD `print.tone` command to allow the user to both view and configure a setting.

```
1→ <wml>
2→ <display>
3→   <card id="main" title="" ontimer="#main" alerts="on">
4→     <timer value="50"></timer>
5→     <p>Baud: $(comm.baud)</p>
6→     <br/>
7→     <p>AP: $(wlan.essid)</p>
8→     <p> </p><br/>
9→     <p> </p><br/>
10→    <p>   <a href="#system">Firmware</a></p>
11→  </card>
12→  <card id="system" title="" ontimer="#main" alerts="on">
13→    <timer value="50"></timer>
14→    <p>Firmware:</p><br/>
15→    <p>$(appl.name)</p><br/>
16→    <p>ZBI State:</p><br/>
17→    <p>$(zbi.key)</p><br/>
18→    <p><a href="#main">Main</a>           <a href="#darkness">Darkness</a></p>
19→  </card>
20→  <card id="darkness" title="" ontimer="#main" alerts="on">
21→    <timer value="50"></timer>
22→    <p>Current: $(print.tone)</p><br/>
23→    <p>Change: </p><do type="accept" label="Up"><setvar name="print.tone"
24→    value="+1.0"/></do><br/>
25→    <p>Change: </p><do type="accept" label="Down"><setvar
26→    name="print.tone" value="-1.0"/></do><br/>
27→    <p> </p><br/>
28→    <p><a href="#main">Main</a>           <a href="#config">Config</a></p>
29→  <card id="config" title="" ontimer="#main" alerts="on">
30→    <timer value="50"></timer>
31→    <p>Printing </p><br/>
32→    <p> Config Label...</p><br/>
33→    <p></p><br/>
34→    <p>Please wait...</p><br/>
35→    <setvar name="file.run" value="e:config.nrd"/>
36→  </card>
37→ </display>
38→ </wml>
```

23	<pre>&lt;p&gt;Change: &lt;/p&gt;</pre> <ul style="list-style-type: none"> <li>• Defines the fixed text "Change: "</li> <li> <pre>&lt;do type="accept" label="Up"&gt;&lt;setvar name="print.tone" value="+1.0"/&gt;&lt;/do&gt;&lt;br/&gt;</pre> </li> <li>• Defines selecting the word "Up" as equal to sending the value "+1.0" for the SGD command "print.tone". In this case, this increases the setting by 1.0.</li> </ul>
24	<pre>&lt;p&gt;Change: &lt;/p&gt;</pre> <ul style="list-style-type: none"> <li>• Defines the fixed text "Change: "</li> <li> <pre>do type="accept" label="Down"&gt;&lt;setvar name="print.tone" value="-1.0"/&gt;&lt;/do&gt;&lt;br/&gt;</pre> </li> <li>• Defines selecting the word "Down" as equal to sending the value "-1.0" for the SGD command "print.tone". In this case, this decreases the setting by 1.0.</li> </ul>

In use, these WML menus look similar to this:



**Note** • GX series printers can display four lines of text. If you are using a GX series printer, remove one line of text from each “card” to use this example.



## Troubleshooting Scenarios

Problem Scenario	Corrective Actions
<p>I loaded a WML menu structure on the printer, but the Factory menu structure is displaying.</p>	<ul style="list-style-type: none"> <li>• "The WML files may have syntax errors. Reconfirm that the correct syntax has been used. When creating WML files it is recommended to start with a simple structure, validate that it's functional and build additional content onto the "known good" example</li> <li>• "Power cycle the printer and watch the start-up sequence - if a "WML ERROR" message displays during the start-up, the index.wml file has a syntax error that needs to be corrected.</li> <li>• "The index.wml file may not have been successfully transferred to the printer. Use a terminal emulation program and the following command to retrieve the index.wml file to the PC for examination: ! U1 setvar "file.type" "E:INDEX.WML"</li> <li>• "The index.wml file may not be present in the E: memory location. Validate that the file is present and correctly named.</li> <li>• "Confirm that straight quotes were used in all instances where the quote character was used (use the " character - not " or ?). SGD commands require the use of the straight quote.</li> <li>• "Confirm that the "WML Menu Cancel" buttons, (Setup and Cancel or Cancel and Setup/Exit or Select) were not held down during start up. These actions will cause the standard menu to display</li> </ul>
<p>Some characters in the menu are cut off or some lines are not displaying at all.</p>	<ul style="list-style-type: none"> <li>• Characters that extend past the width of the display are truncated, reposition the field as needed.</li> <li>• Check that you have not exceeded that maximum number of lines the display allows (5 lines on ZM and Xi4 series units, 4 lines on the GX series).</li> </ul>
<p>My WML menu structure is displaying, but one of the "cards" is not displaying or is unreachable.</p>	<ul style="list-style-type: none"> <li>• The missing "card" may not have been linked to from any of the visible "cards". Review you WML content to ensure that the correct links exist.</li> <li>• The WML files may have syntax errors, reconfirm that the correct syntax has been used. When creating WML files it is recommended to start with a simple structure, validate that it's functional and build additional content onto a known good example.</li> <li>• If the missing card content is contained in a separate .wml file, confirm that the necessary .wml files have been transferred to the printer.</li> </ul>

Problem Scenario	Corrective Actions
<p>My WML menu structure uses SGD commands to display current settings, but the settings are not displaying.</p>	<ul style="list-style-type: none"> <li>• Validate that the correct syntax was used for the SGD command.</li> <li>• Check the manual page for the command being used. Confirm that the command is supported by the printer &amp; firmware. Use a terminal emulation program to send just the command being used to validate it functions outside the WML menu structure.</li> <li>• Confirm that straight quotes were used in all instances where the quote character was used (use the " character – not “ or ”). SGD commands require the use of the straight quote.</li> <li>• Characters that extend past the width of the display are truncated, reposition the field as needed.</li> <li>• Check that you have not exceeded that maximum number of lines the display allows (5 on ZM and Xi4 series units).</li> </ul>
<p>My WML menu structure used SGD commands to allow the user to alter printer settings, but the settings are not getting changed as expected.</p>	<ul style="list-style-type: none"> <li>• Validate that the value being used in the <code>value=</code> parameter of the <code>&lt;do&gt;</code> tag is supported by the SGD command.</li> <li>• Validate that the correct syntax was used for the SGD command.</li> <li>• Check the manual page for the command being used. Confirm that the command is supported by the printer &amp; firmware. Use a terminal emulation program to send just the command being used to validate it functions outside the WML menu structure.</li> <li>• Confirm that straight quotes were used in all instances where the quote character was used (use the " character – not “ or ?). SGD commands require the use of the straight quote.</li> </ul>
<p>My WML menu structure uses .nrd files to send commands to the printers ZPL or SGD engine, but the commands don't seem to be getting sent when the user selects the on-screen link for the action.</p>	<ul style="list-style-type: none"> <li>• Confirm that the .nrd files are present in E: memory and named as expected. Resend or rename the files if necessary.</li> <li>• Confirm that the WML menu structure is using the correct file name(s).</li> <li>• Confirm that the commands in the files work as expected, independently of the WML menu or .nrd file.</li> <li>• Validate that ZPL and SGD commands have not been interlaced.</li> </ul>
<p>I am using the <code>CISDFCRC16</code> command to transfer files, but the files are either not being transferred to the printer or are showing up with a zero (0) byte size.</p>	<ul style="list-style-type: none"> <li>• Confirm that the Hexadecimal value used for the File Size parameter is correct. This value must be an eight digit file size specified in hexadecimal which indicates the number of bytes in the <code>&lt;data&gt;</code> section of the command. See the full manual page on the <code>CISDFCRC16</code> command for additional details.</li> <li>• Validate that the CRC and Checksum parameters are correct (using the "0000" value for these parameters is recommended).</li> <li>• Confirm that the WML file name and extension are in upper case characters.</li> <li>• Confirm that the exclamation mark (!) was included before the command name ("! <code>CISDFCRC16</code>").</li> </ul>