

AllMyPapers

X9 RCCTM

**System for creating ICL files
from RCC generated images**

User Manual

Version 2.2



© 2015 AllMyPapers. All Rights Reserved.

Printed in the United States of America.

X9 RCC is a trademark of AllMyPapers. Microsoft Windows are trademarks of Microsoft Corporation. All other brand and product names are trademarks or registered trademarks of their respective companies.

All My Papers
13750 Serraoaks
Saratoga, CA 95070
(408) 366-6400
www.allmypapers.com

Table of Contents

1	<i>Introduction</i>	<i>1</i>
1.1	X9 RCC Overview.....	1
2	<i>Configuration File.....</i>	<i>4</i>
3	<i>Payment File</i>	<i>11</i>
4	<i>Check Template Design</i>	<i>12</i>
5	<i>Platform Requirements</i>	<i>13</i>
5.1	Hardware Requirements.....	13
5.2	Software Requirements	13
6	<i>Installation</i>	<i>13</i>
6.1	X9 RCC Runtime License	13
6.2	X9RCC Installation	13
6.2.1	Files and Directories	13
7	<i>Setup.....</i>	<i>15</i>
7.1	Output Options	15
7.2	Application Locations.....	17
7.3	PrintBatch Configuration	18
8	<i>Operations</i>	<i>19</i>
8.1	User Interface Operations	19
8.2	View Menu.....	21
8.3	Automatic Operations	22
9	<i>Microsoft Document Imaging Setup</i>	<i>22</i>

1 INTRODUCTION

1.1 X9 RCC Overview

Remotely created checks (RCC) or demand drafts are checks that are created by or on behalf of the depositor and authorized by the check account holder. They are commonly used to process preauthorized remittance payments. With the advent of Check 21 checks can now be truncated with the check image and MICR data cleared in X9.37 Image Cash letter files. All My Papers (AMP) X9 RCC System will perform a paperless generation of X9.37 Image Cash Letters containing RCC check images. The X9 RCC system is the integration of proprietary and exclusive AMP Software Development Toolkits. AMP's AmpLib SDK is used to generate RCC check images from text data. X9Lib SDK is used to bundle the RCC images into Image Cash Letter file formats compliant with image exchange or deposit formats.

Figure 1 shows the architecture X9 RCC that is built upon AMP's Software Development Kits (SDK). This is an automated solution that will import the payment file that contains all the check payment information details and then produce an X9.37 Image Cash Letter (ICL) that conforms to the requirements for clearing through the Federal Reserve Bank's (FRB) clearing system. Additionally ICL files can be formatted for deposit into commercial banks. AMP technology supports deposit ICL formats for the majority of the large banks.

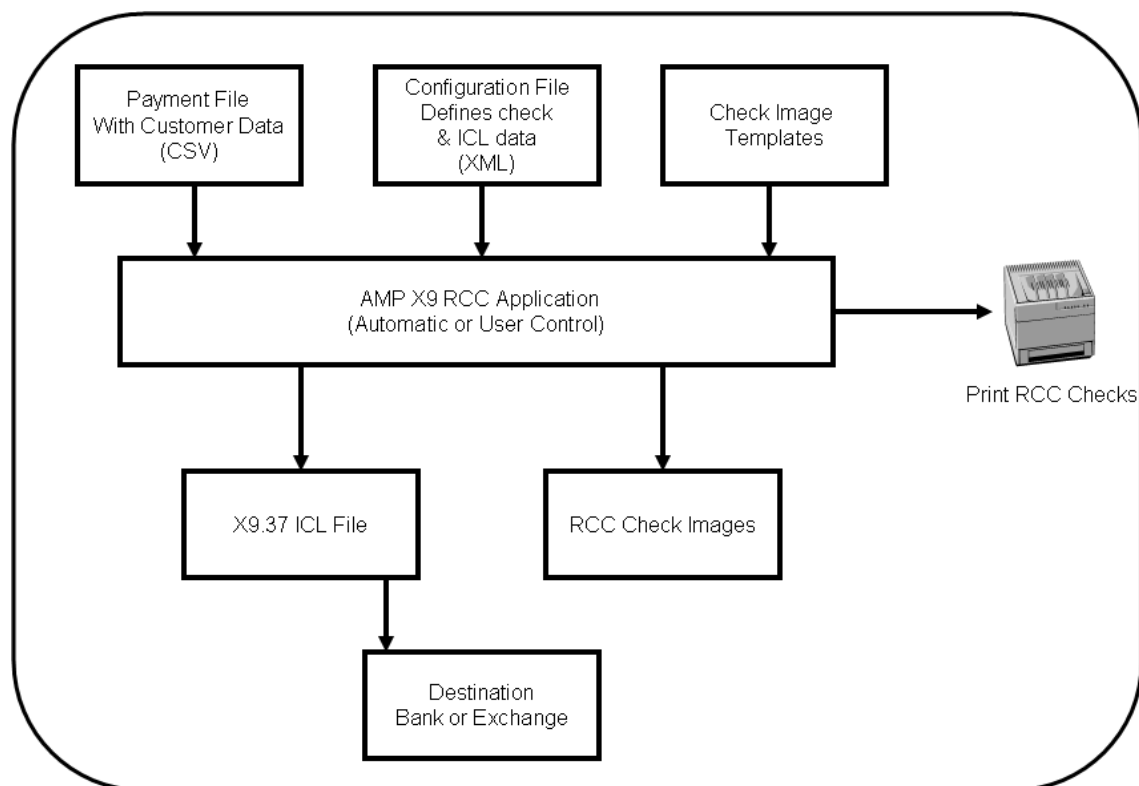


Figure 1 RCC ICL File Generation System

Features of this system include:

- Supports custom design of check images
- Manual or automated lights out operation
- 100% configurable. No programming required
- Supports dozens of commercial bank deposit formats
- Capacity to generate over 40,000 RCC items in a single ICL file.
- Generate items at over 1000 Items per minute
- RCC images can be automatically printed if source document is required
- Fraud deterrent feature. Depositor's name and account number is required to be endorsed on check images.

To produce RCC check images the application imports an XML Configuration file, a text payment file, and a front and back check image template file. The configuration file defines the following:

- Filenames and locations of the check image template files
- Check image zone definitions:
 - content
 - font
 - Format
 - location
 - orientation
- Output ICL format
- MICR Code line formats
- Payment file contents

The system first generates the RCC check images. An option allows the automatic printing of these check images if the source document is required for any purposes. The RCC check images are then bundled into a X9.37 ICL format compatible for image clearing through the exchange networks. The application can then convert this to a commercial bank deposit format.

The check image templates can be created with MS Word or any other document or image editing application. This document can be customized according to the customer's requirements. Layout and graphic elements can be designed as long as they conform to industry check design guidelines. Fields are defined for the payment data. An E13B MICR font is provided to encode the MICR code line fields.

2 CONFIGURATION FILE

The configuration file defines the format and contents of the resultant RCC check images and the format of the output ICL file. It is an XML format that can be edited with a text editor or an XML editor. It contains element sections as follows:

Element	RCC_PAY_CFG
Description	Defines the document type
Attribute	Description
documenttype	<p>Defines the set of rules that the application will use to create the RCC document. Valid values are:</p> <p>CHECK – for creating check images</p> <p>DEPOSIT – For creating deposit ticket images</p> <p>Currently only a value of “CHECK” is supported</p>

Example:

```
<RCC_PAY_CFG documenttype="CHECK"> <!--valid values are CHECK or DEPOSIT-->
```

Element	image_templates
Description	Defines the filenames and locations of the document image templates
Attribute	Description
front	Path and filename of the front image template
back	Path and filename of the back image template

Example:

```
<image_templates>
  <img_template front="C:\Program Files\AllMyPapers\X9 RCC\Templates\default_front.tif"></img_template>
  <img_template back="C:\Program Files\AllMyPapers\X9 RCC\Templates\default_back.tif"></img_template>
</image_templates>
```

Element	ACCOUNT_CHECKNUM_Definitions
Description	<p>Defines the format of the ON-US and AUX-ON-US MICR fields</p> <p>“C” – Check Number</p> <p>“A” – Account Number</p> <p>“/” – ON-US symbol (“#”)</p> <p>“-” – dash (“ ”)</p>

	<p>“ “ – SPACE</p> <p>Multiple formats can be defined. Each will be selectable by the index number</p>
Attribute	Description
index	An incrementing number for each definition. This is the number that would appear in the payment file to select the required format
type	<p>Document Type:</p> <p>CORPORATE – for corporate check MICR format</p> <p>PERSONAL – for consumer check MICR format</p>
auxonus	<p>Defines the format for the AUX-ON-US field</p> <p>Examples:</p> <p>“/C” – Produces – ⑈ 0 1 2 3 4 5 ⑈</p> <p>“” – Produces nothing</p>
onus	<p>Defines the format for the ON-US field</p> <p>Examples:</p> <p>“A/C” – Produces – 1 2 3 4 5 6 7 8 ⑈ 0 1 2 3</p> <p>“C/A” – Produces – 0 1 2 3 ⑈ 1 2 3 4 5 6 7 8 ⑈</p> <p>“C-A” – Produces – 0 1 2 3 ⑈ 1 2 3 4 5 6 7 8 ⑈</p>

Example:

```
<ACCOUNT_CHECKNUM Definitions>
  <format index="1" type="CORPORATE" auxonus="/" onus="A"/>
  <format index="2" type="PERSONAL" auxonus="" onus="A/C"/>
  <format index="3" type="PERSONAL" auxonus="" onus="C/A"/>
</ACCOUNT_CHECKNUM Definitions>
```

Element	param_definitions
Description	<p>This section defines all the parameters that will be used to create the RCC images. A parameter can be defined and set in the configuration file, or its value can come from the payment file.</p> <p>For documents defined as “CHECK” here are the parameter rules:</p> <p>There are some predefined parameters that the application is expecting. RCC images will not be generated if the value of these parameters is not set in the configuration file or provided in the payment file. The required parameters are:</p> <ul style="list-style-type: none"> • payee_name • payee_account_nbr • payment_amt • check_nbr

	<ul style="list-style-type: none"> • check_type (select definition type for Account and Check Number formats) • bank_route – Payor Bank Routing number • account_nbr – Payor account number • reference_nbr - check payment reference number used as ECE Sequence <p>The following parameters values are generated by the application:</p> <ul style="list-style-type: none"> • MICR_ROUTE • MICR_AUX • MICR_ONUS (MICR codeline) • legal_amount
Attribute	Description
param	Name of the parameter
cfg_file_pos	Position in the payment file (1,2,3...) Payment file is a comma separated value file
length_MIN	The minimum expected string length of the value provided in the payment file. Used in validation.
length_MAX	The maximum expected string length of the value provided in the payment file. Used in validation.
required	Indicates if the parameter is required. If missing the application will terminate with an error condition. Values are “True” or “False”
value	Sets the value of the parameter. A parameter defined with a value will not exist in the payment file.

Example:

```

<param_definitions>
  <REQUIRED>
    <define param="payee_name"          cfg_file_pos="1"  length_MIN="5" length_MAX="30" required="true"/>
    <define param="payee_account_nbr"    cfg_file_pos="2"  length_MIN="5" length_MAX="17" required="true"/>
    <define param="reference_nbr"        cfg_file_pos="3"  length_MIN="15" length_MAX="15" required="true"/>
    <define param="payment_amount"       cfg_file_pos="4"  length_MIN="1" length_MAX="11" required="true"/>
    <define param="check_nbr"           cfg_file_pos="6"  length_MIN="1" length_MAX="10" required="true"/>
    <define param="check_type"           cfg_file_pos="7"  length_MIN="1" length_MAX="1"  required="true"/>
    <define param="account_nbr"          cfg_file_pos="19" length_MIN="4" length_MAX="17" required="true"/>
    <define param="bank_route"           cfg_file_pos="20" length_MIN="9" length_MAX="9"  required="true"/>
  </REQUIRED>

  <DEFINED>
    <define param="payment_date"          cfg_file_pos="5"  length_MIN="6" length_MAX="22" required="true"/>
    <define param="account_name1"         cfg_file_pos="8"  length_MIN="1" length_MAX="20" required="true"/>
    <define param="account_name2"         cfg_file_pos="9"  length_MIN="1" length_MAX="20" required="false"/>
    <define param="account_address1"      cfg_file_pos="10" length_MIN="1" length_MAX="20" required="true"/>
    <define param="account_address2"      cfg_file_pos="11" length_MIN="1" length_MAX="20" required="false"/>
    <define param="account_city"          cfg_file_pos="12" length_MIN="2" length_MAX="20" required="true"/>
    <define param="account_state"         cfg_file_pos="13" length_MIN="2" length_MAX="2"  required="true"/>
    <define param="account_zip"           cfg_file_pos="14" length_MIN="5" length_MAX="10" required="true"/>
    <define param="bank_name"             cfg_file_pos="15" length_MIN="2" length_MAX="30" required="true"/>
    <define param="bank_city"             cfg_file_pos="16" length_MIN="2" length_MAX="20" required="true"/>
    <define param="bank_state"            cfg_file_pos="17" length_MIN="2" length_MAX="2"  required="true"/>
    <define param="bank_zip"              cfg_file_pos="18" length_MIN="5" length_MAX="10" required="true"/>
    <define param="payee_phone"           cfg_file_pos="21" length_MIN="2" length_MAX="12"  required="true"/>
    <define param="BOFD_RT"               value="999999999"/>
  </DEFINED>
</param_definitions>

```

Element	zone_definitions
Description	Defines the zones on the RCC images where information will be printed. Defines the location, content and format of each zone
Sub Element	front
Description	Contains the zones definitions for the front image
Sub Element	rear
Description	Contains the zones definitions for the rear image
Attribute	Description
name	Name of the zone
view	"front" – for zone on front image "back" – for zone on back image
x	Horizontal distance from the left side of the document image to the left side of the zone. Measured in inches. (E.g. 2.25 = 2.25 inches). Accuracy is 0.01 inch.
y	Vertical distance from the top side of the document image to the top side of the zone. Measured in inches. (E.g. 2.25 = 2.25 inches). Accuracy is 0.01 inch.
width	Horizontal width of the zone. Measured in inches.
height	Vertical Height of the zone. Measured in inches.
textlines	Number of text lines that will be printed in the zone

<code>font</code>	Name of the font to be used for the zone or text line.
<code>size</code>	Size of the font in points.
<code>justification</code>	Justification of the text in the zone or line. Values are “left”, “right” or “center”.
<code>orientation</code>	Orientation of the zone. Values are “vertical” or “horizontal”.
<code>bold</code>	Text will be printed in bold or normal. Values are “True” or “false”.
Sub Element	<code>line</code>
Description	Each zone will have 1 or more lines of text. Each line will have a line definition.
Attribute	Description
<code>string</code>	<p>This is the string contents of the line. The line can contain from 0 to 9 variables whose values come from the parameters. To add a variable use “%n” where n is the index number of the variable. The first variable will be %1, 2nd will be %2, etc.</p> <p>Examples:</p> <pre><line string="%1" var1="payee_name"/></pre> <p>will print the string value of the payee_name.</p> <pre><line string="Void after 180 days" size="8"/></pre> <p>will print “Void after 180 days” in 8 point font.</p>
<code>var1</code>	Defines the parameter to use for variable 1. Use var2, var3, ... for multiple variables.
<code>format1</code>	<p>Supports optional formats for date and amount fields.</p> <p>amount format values:</p> <ul style="list-style-type: none"> - amount1 199.99 - amount2 \$199.99 - amount3 *****199.99***** - amount4 \$****199.99***** <p>date format values:</p> <ul style="list-style-type: none"> - date1 MM/DD/YYYY - date2 MMMMM DD YYYY
<code>Line format options</code>	<p>Additionally for each line you may redefine:</p> <ul style="list-style-type: none"> • font • size • justification • orientation • bold

Example:

```

<zone_definitions>
  <front>
    <!--Signature field definition-->
    <zone name="signature" view="front" x="5.5" y="1.9" width="2" height="0.5" textlines="4"
      font="MS Reference Sans Serif" size="10"
      justification="center" orientation="horizontal" bold="false">
      <line string="Void after 180 days" size="8"/>
      <line string="Signature on file" size="11"/>
      <line string="This check has been authorized"/>
      <line string="by your depositor"/>
    </zone>

    <!--Legal Amount definition-->
    <zone name="legal_amount" view="front" x="1" y="1.35" width="3" height="0.25" textlines="1"
      font="MS Reference Sans Serif" size="10"
      justification="left" orientation="horizontal" bold="false">
      <line string="****%1" var1="legal_amount"/>
    </zone>

    <!--MICR codeline definition-->
    <zone name="MICR_RT" view="front" x="2.45" y="2.55" width="1.5" height="0.2" textlines="1"
      font="MICR" size="12"
      justification="left" orientation="horizontal" bold="false">
      <line string="%1" var1="MICR_ROUTE"/>
    </zone>

```

Element	ICL_definitions	
Description	Defines the contents and format of the output ICL file.	
Sub Element	x937	
Description	Defines origin and destination header record contents	
Attribute	Description	
Destination_routing_number	9 digit destination institution routing number	
Destination_name	Destination institution name (18 digits maximum)	
Origin_routing_number	9 digit origin institution routing number	
Origin_name	Origin institution name (18 digits maximum)	
Test_File	"True" for test file, "False" for production file	
Destination_Format	A two digit number that defines the commercial bank deposit format that will be produced. Contact AMP for more information.	
Delete_Records	Removes records from the output ICL file. Some deposit formats require some records to not be present.	
CreditFrontImage	Defines the filename and path of the front image that will be used for a credit item.	

<code>CreditBackImage</code>	Defines the filename and path of the back image that will be used for a credit item.
<code>Update_Credit_Image</code>	<p>“True” – Annotate the credit image with deposit information</p> <p>“False” – Do not annotate the credit image (default)</p>
<code>Delete_Records</code>	<p>Record numbers to delete (usually Records 54, 28, 26)</p> <p>Comma separated</p>
<code>FIRoutingNumber</code>	For Canadian deposit ICL formats this is the actual routing number of the depositing Institution. Can be a Canadian or US Financial Institution.
<code>Summary_Item_Include_Credit_Item</code>	<p>ICL Control Item Totals to include Credit Items (True/False)</p> <p>Default is “True”</p>
<code>Summary_Amount_Include_Credit_Item</code>	<p>ICL Control Amount Totals to include Credit Items (True/False)</p> <p>Default is “True”</p>
Element	<code>setRecordField</code>
Description	Sets the value of any text field in any record in an ICL file.
Attribute	Description
<code>record</code>	Record number
<code>field</code>	Field number
<code>all</code>	<p>“True” – Set the field in all Records of this type.</p> <p>“False” – Set field in only the first Record of this type.</p>
<code>value</code>	The value to set.
<code>afterconversion</code>	<p>“True” – Set the value after conversion to Deposit format.</p> <p>“False” – Set the value before conversion to Deposit format.</p> <p>This is optional. Default is “False”</p>
Element	<code>setOneRecordField</code>
Description	Sets the value of any text field in any record in an ICL file. Allows for selecting specific items/records in specific locations
Attribute	Description
<code>record</code>	Record number
<code>field</code>	Field number

cashletter	Cash letter position in file
bundle	Bundle location within the cash letter
item	Item location within the bundle
value	The value to set. The esc sequences of: \$DATE – replaces with the date in format YYMMDD \$TIME – replaces with the time in HHMMSS format
afterconversion	“True” – Set the value after conversion to Deposit format. “False” – Set the value before conversion to Deposit format. This is optional. Default is “False”

Example:

```
<ICL_definitions>
  <x937 Destination_routing_number ="123456780"/>
  <x937 Destination_name ="FRBanydistrict"/>
  <x937 Origin_routing_number="012345672"/>
  <x937 Origin_name ="RCCCreator"/>
  <x937 Test_File ="True"/>
  <x937 Destination_Format="20"/>
  <x937 Delete_Records="26,54"/>
  <x937 CreditFrontImage="c:\front.tif"/>
  <x937 CreditBackImage="c:\back.tif"/>
  <setRecordField record="10" field="12" all="False" value="1112223333"/>
  <setRecordField record="99" field="7" all="False" value="1112224444"/>
  <setRecordField record="52" field="4" all="True" value="10"/>
</ICL_definitions>
```

3 PAYMENT FILE

The Payment file contains the data that will be printed on to the checks. The contents are defined in the configuration file. This file is a text file with each parameter value separated by a comma (CSV format). Each value must also be delimited by double quotes. There is one line for every RCC item that is to be generated. Below is an example of one line from a payment file.

```
"Check Payee 1","1234567890321654","8888888888888888","1.0000","3/2/2010 12:00:00
AM","9876","2","Abby Davis","","6548 Sterling Circle","","Lunar","WI","56789","Third Credit
Union","Lunar","WI","56789","5555444400","069327216","800-328-7448"
```

4 CHECK TEMPLATE DESIGN

The application requires a front and back image template for the documents you will be generating images for.

The check template is used by the application to create the RCC images by adding the information defined in the configuration file. A check template can be created in MS Word by defining a custom document size (8.5 X 3.0) for the check images. Boxes, lines, text can be added as required. To create an image file you will need to redirect the printer output to Microsoft Document Imaging that has been configured to produce TIFF images at 200 DPI resolution. Refer to section 9 for details on setting up Microsoft Document Imaging.

Figure 2 and Figure 3 show the check template design.

DATE	CHECK NO.	AMOUNT

PAY
TO THE
ORDER
OF

Figure 2 Check Template - Front View

DO NOT WRITE OR STAMP BELOW THIS LINE

Figure 3 Check Template - Rear View

5 PLATFORM REQUIREMENTS

5.1 Hardware Requirements

This section describes the minimum required hardware platform to operate the X9 RCC system achieving the performance objectives. The customer is responsible for providing this hardware platform. This is based on a typical volume of 500 items per day, with a peak volume of 2000 items per day.

Windows PC/Server

- Single Processor, 2 GHz
- 1 GB RAM
- 60 GB Hard drive for Operating System and Application

5.2 Software Requirements

This section describes the software platform requirements. The customer is responsible for providing the platform environment.

Operating System: Windows (XP, Server 2008-12, Windows 7/8)

6 INSTALLATION

6.1 X9 RCC Runtime License

AMP will provide a set of license codes that will need to be installed on the operating platform. This requires the AMP License Manager to be downloaded and installed from:

http://www.allmypapers.com/license_manager.htm.

6.2 X9RCC Installation

This is a self installing executable that is part of the deliveries.

6.2.1 Files and Directories

The X9 RCC program files will be installed in the directory structure as shown in Figure 4. AMP will provide a self installing executable that will create the required directories and install the files.

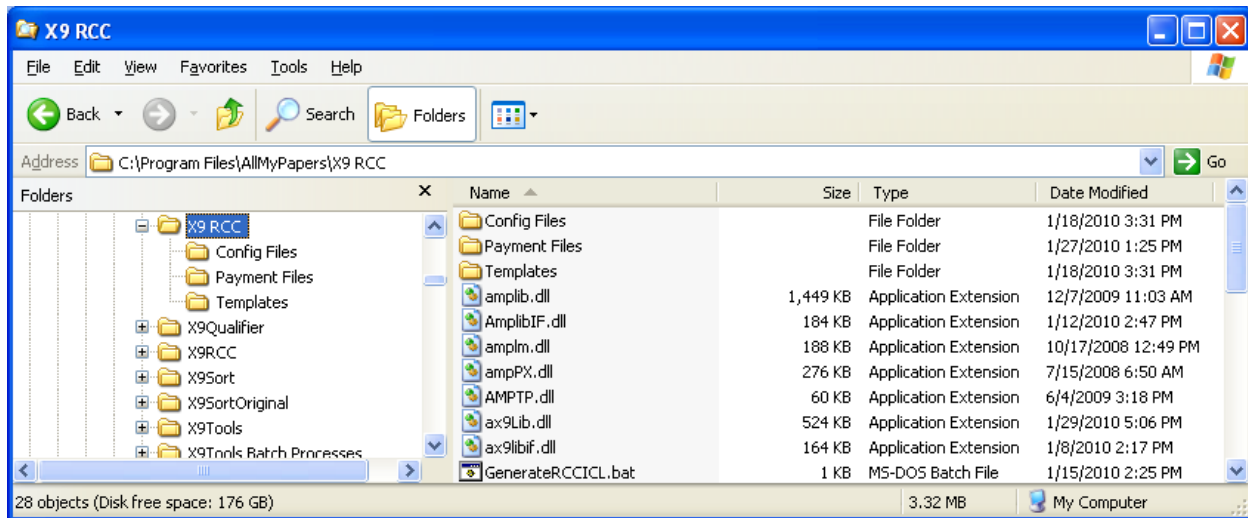


Figure 4 X9 RCC Installation Directories

The directories are for the following files:

X9RCC – Contains the application and runtime files

Config Files – Contains a sample configuration file.

Payment Files – Contains a sample payment file.

Templates – Contains sample check image and deposit ticket templates

Help – Contains the Users manual

Additionally the following files will be installed on the file system:

- MICR font file "MICR.ttf" will be installed in "C:\WINDOWS\Fonts". This enables MS Word to print the MICR code line on the check images

The application will create additional directories to operate. A working directory called X9RCC will be created in the application data directory. See Figure 5 below.

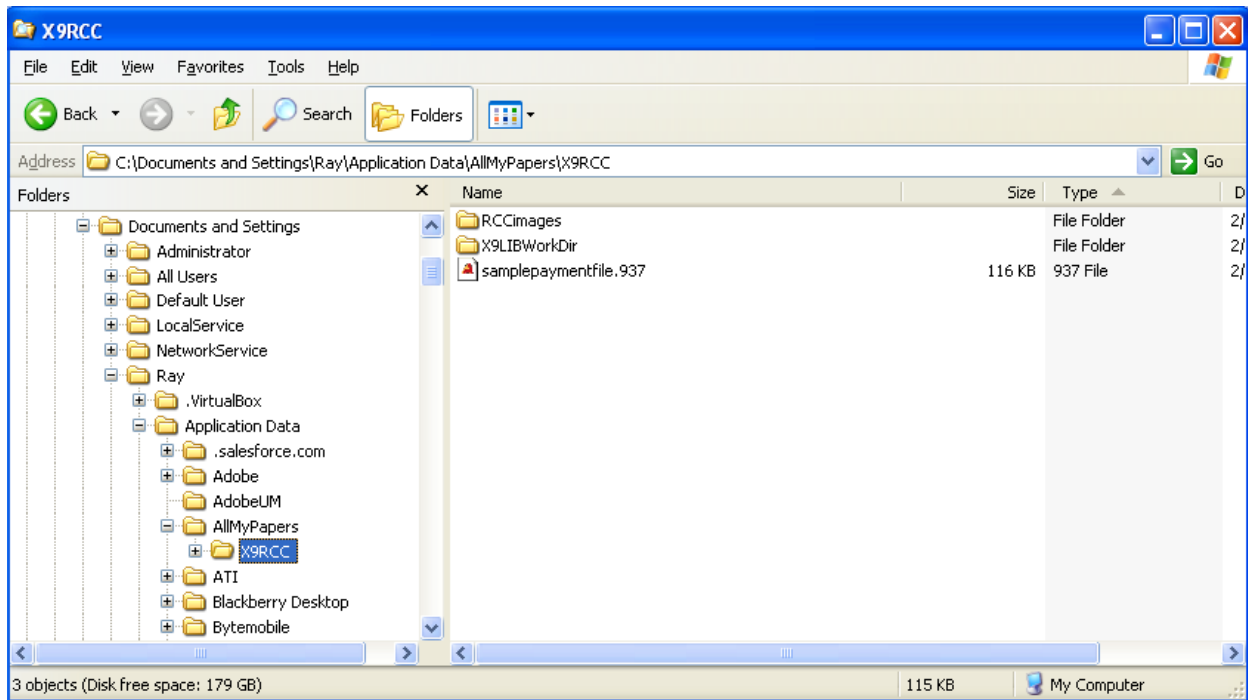


Figure 5 Working Directories

The directories are for the following purposes:

X9RCC – The output ICL file will be put here by default unless specified differently. Intermediate files are also created here. The application will also create a log file here.

RCCimages – The generated RCC images will be generated in this directory unless specified differently.

X9LIBWorkDir – This is a directory the application uses to process. If anything goes wrong there are log files available here that can be used to diagnose the issues.

7 SETUP

This section describes the additional setup procedures that can be performed after installation to configure the desired operation

7.1 Output Options

With the application started select the menu item Configuration – Output Options

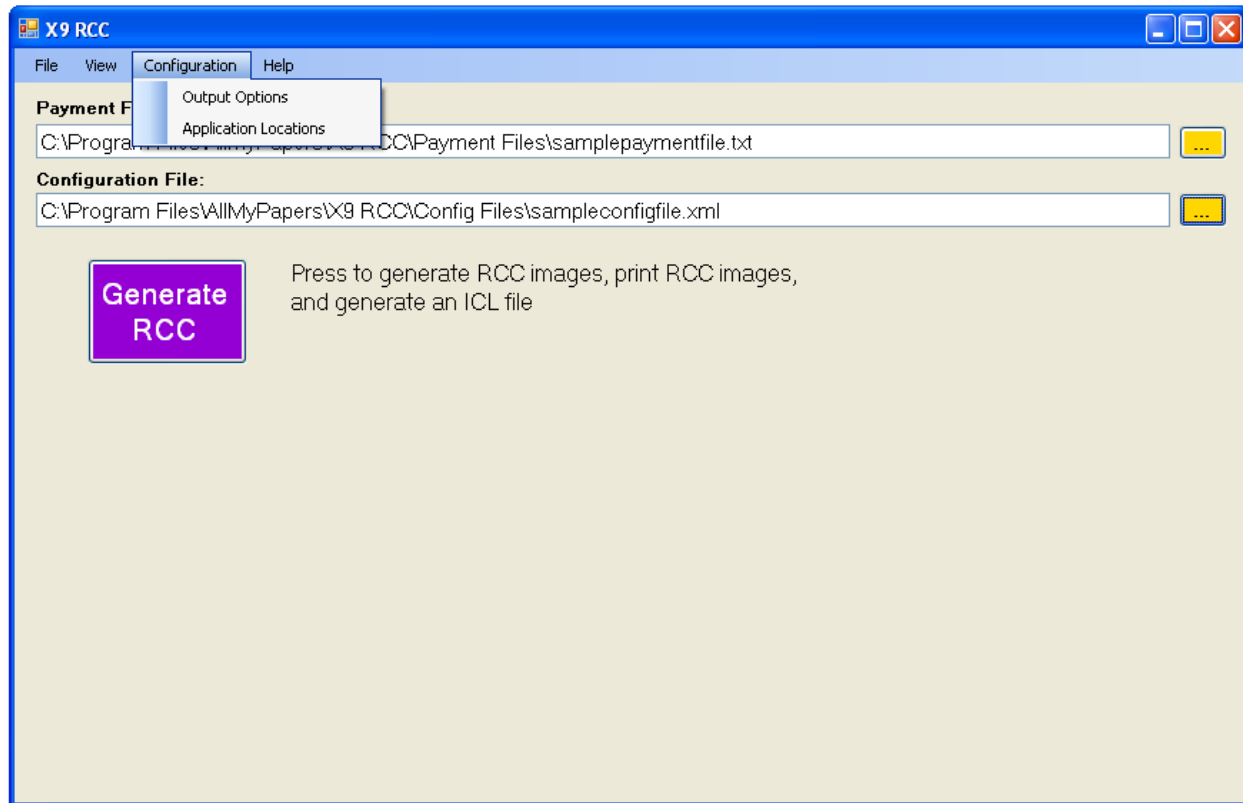
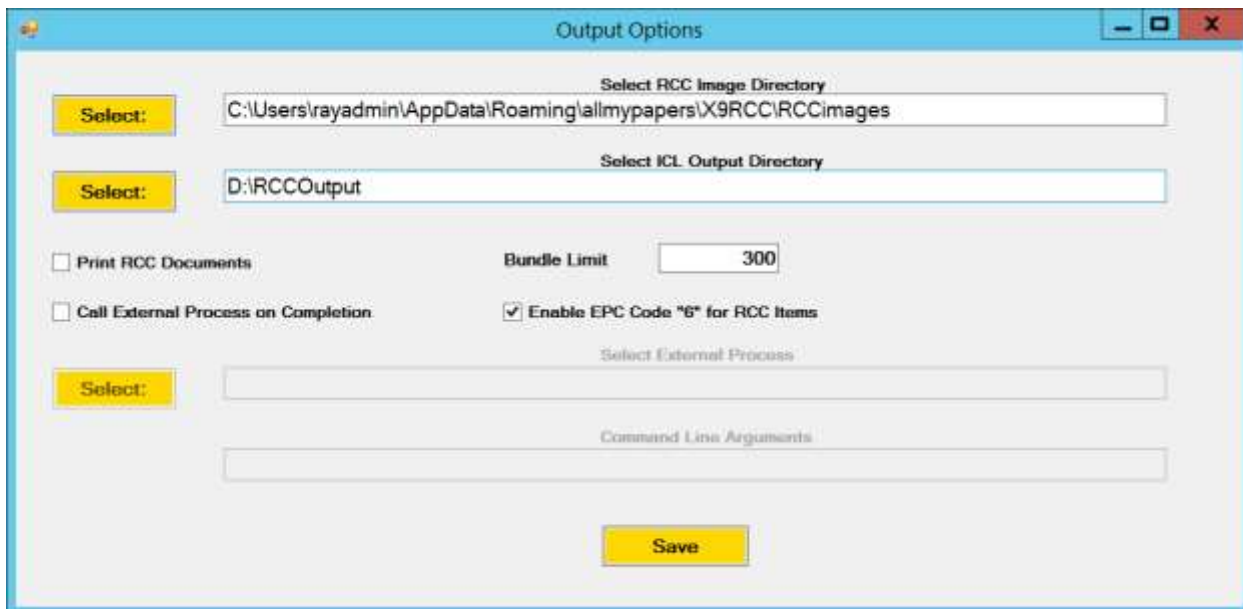


Figure 6 Configuration Menu

The Output Options screen will be displayed



Select RCC Image Directory – This is the directory location that the application will generate the RCC check images. Select a new directory if you do not want to use the default directory. The application will **erase the images prior to each new job**.

Select ICL Output Directory - This is the directory location that the application will create the final ICL file. **Select a new** directory if you do not want to use the default directory.

Bundle Limit – Select a bundle limit if you want the application to control the maximum number of items that can be placed in a single bundle. Value must be from 1 to 9,999. Leave blank if you do not want to limit bundles.

Print RCC Documents – Select this if you want the application to automatically print the RCC document images.

Call External Process on Completion – Select this if you want to invoke another process upon completion of the generation of the ICL file.

Select External Process – Use this to select the external process to be invoked. This process will be called with the ICL pathname as the first argument

Command Line Arguments – You can add additional arguments if required for the external process call.

Select “Save” to store the new settings.

Enable EPC Code “6” for RCC Items – In 2015 the FRB requires RCC items to have an EPC code of “6” applied to identify them as RCC items.

7.2 Application Locations

X9 RCC can invoke two additional applications. It can invoke an application to print the RCC images (PrintBatch) and it can invoke the X9 Viewer application to view the contents of the ICL files.

To use these functions requires the installation of:

- XipPrint Application
- X9 Viewer Application

Nothing needs to be done if these applications are installed in the default locations. If they are installed in non standard locations then the application allows you to define the new location.

With the application started select the menu item: Configuration – Application Locations

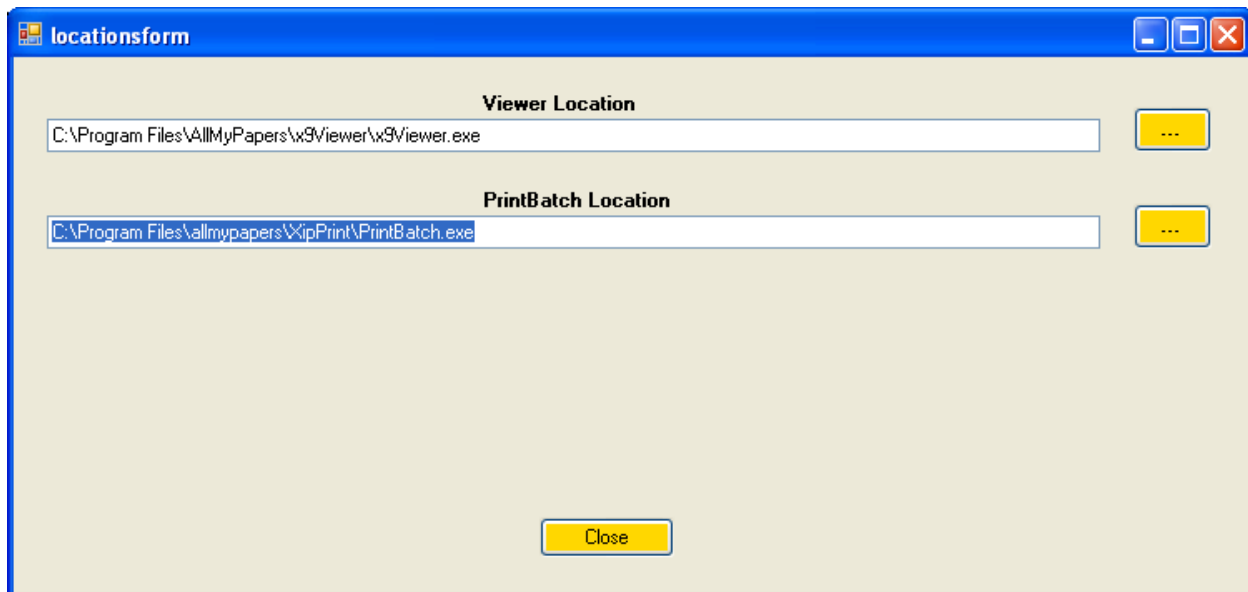


Figure 7 Application Locations Form

Browse to the new application locations and close the form. Close the form and application to save the settings.

7.3 PrintBatch Configuration

To print RCC images from the application requires the configuration of the PrintBatch.exe utility located in the XipPrint application directory. See Figure 8.

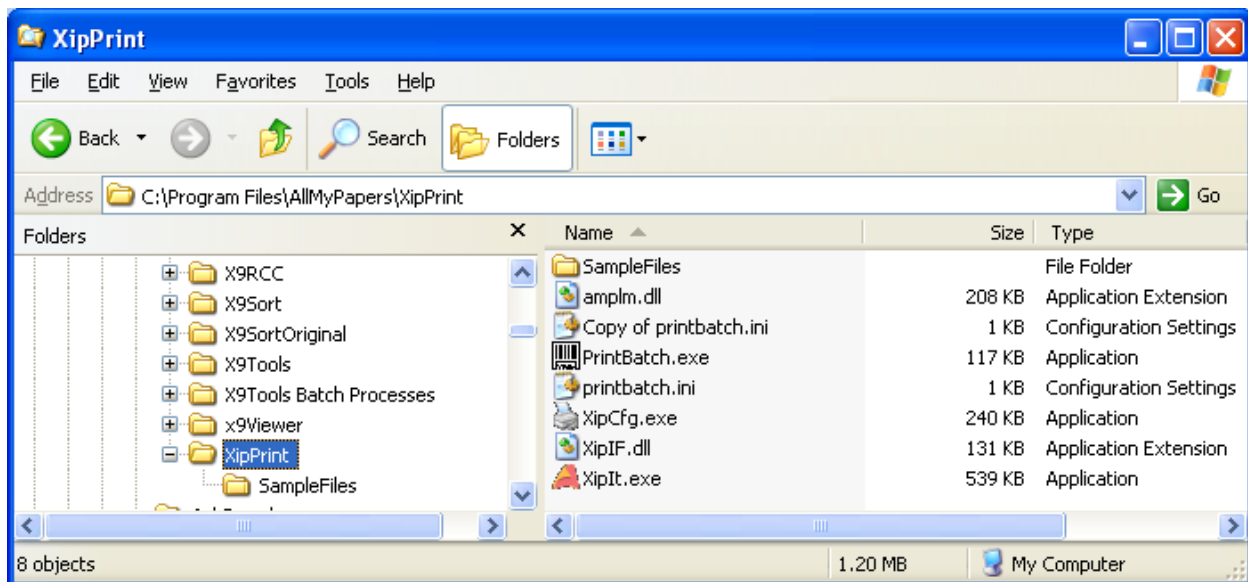


Figure 8 PrintBatch Utility

To use PrintBatch you must first configure XipPrint to use the required printer. Refer to help materials on installing and using XipPrint.

To configure PrintBatch run the utility and select the menu item: File -> Printing Options. Configure the Printing Options form as shown in Figure 9.

Figure 9 Printing Options Form

Select the active printer for your configuration.

8 OPERATIONS

8.1 User Interface Operations

Figure 10 shows the main operation's screen.

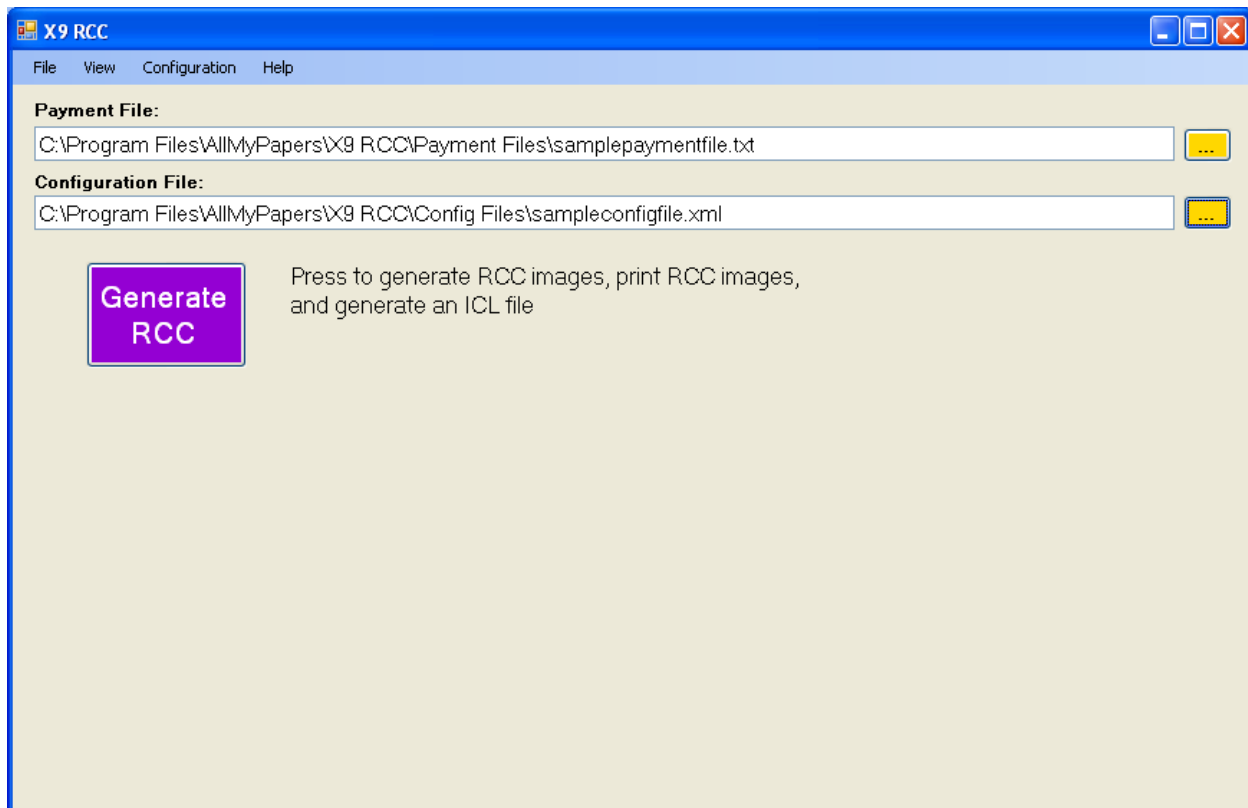


Figure 10 Main Screen

Payment File – Select the payment file that contains the item payment data to create RCC images form.

Configuration File – Select the configuration file that contains the definitions for the RCC images and ICL output file formats.

Generate RCC Button – Select to start the generation of RCC images and creation of the RCC ICL output file.

After processing is complete the screen will look like Figure 11. Progress is shown during processing. The screen will list a summary of the output file created.

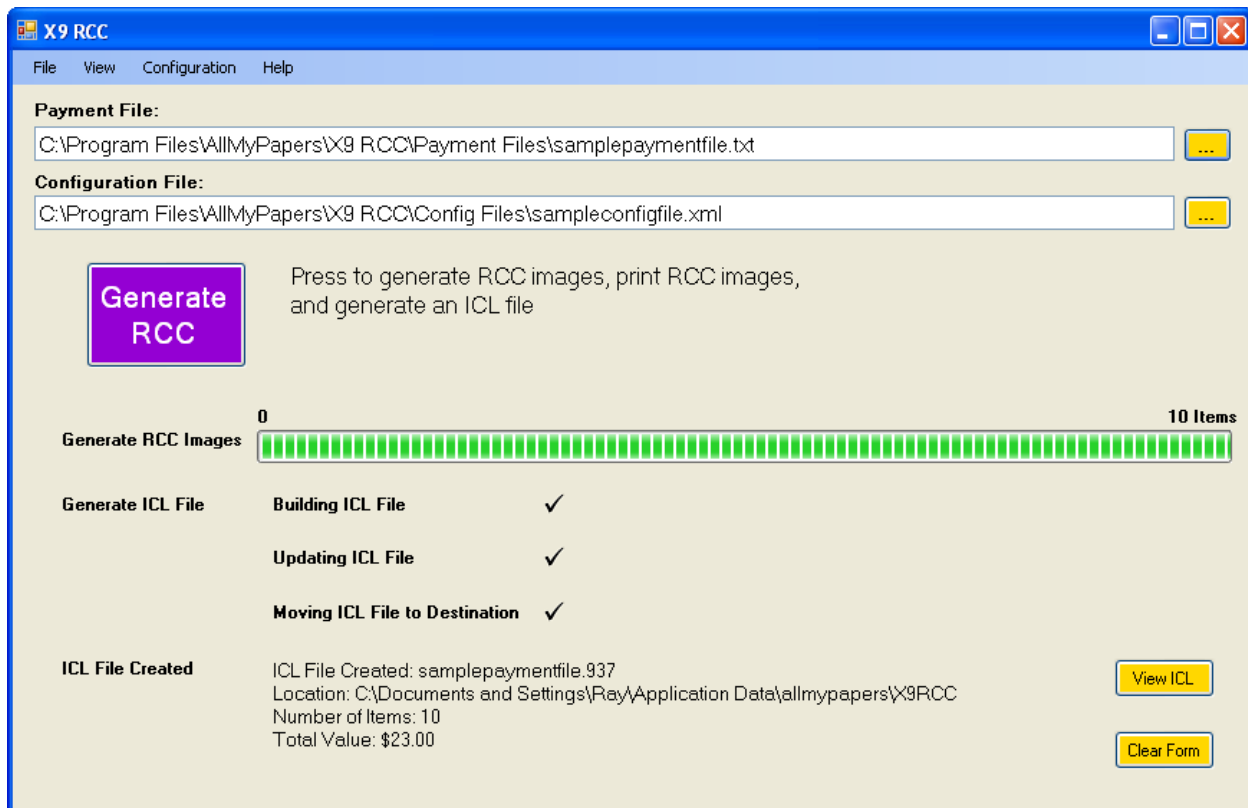


Figure 11 Processing Complete

View ICL – Select to invoke the X9 Viewer to view the output ICL file.

Clear Form – Select to clear this form.

8.2 View Menu

Figure 12 shows the View menu. You have 4 options:

ICL File – View the generated ICL file with the X9 Viewer application.

Configuration File – View/edit the configuration file.

Payment File – View the payment file.

RCC Log File - View the job log file. This will normally contain a summary of the file just generated. If an error occurred during processing then this file will contain additional information on the error.

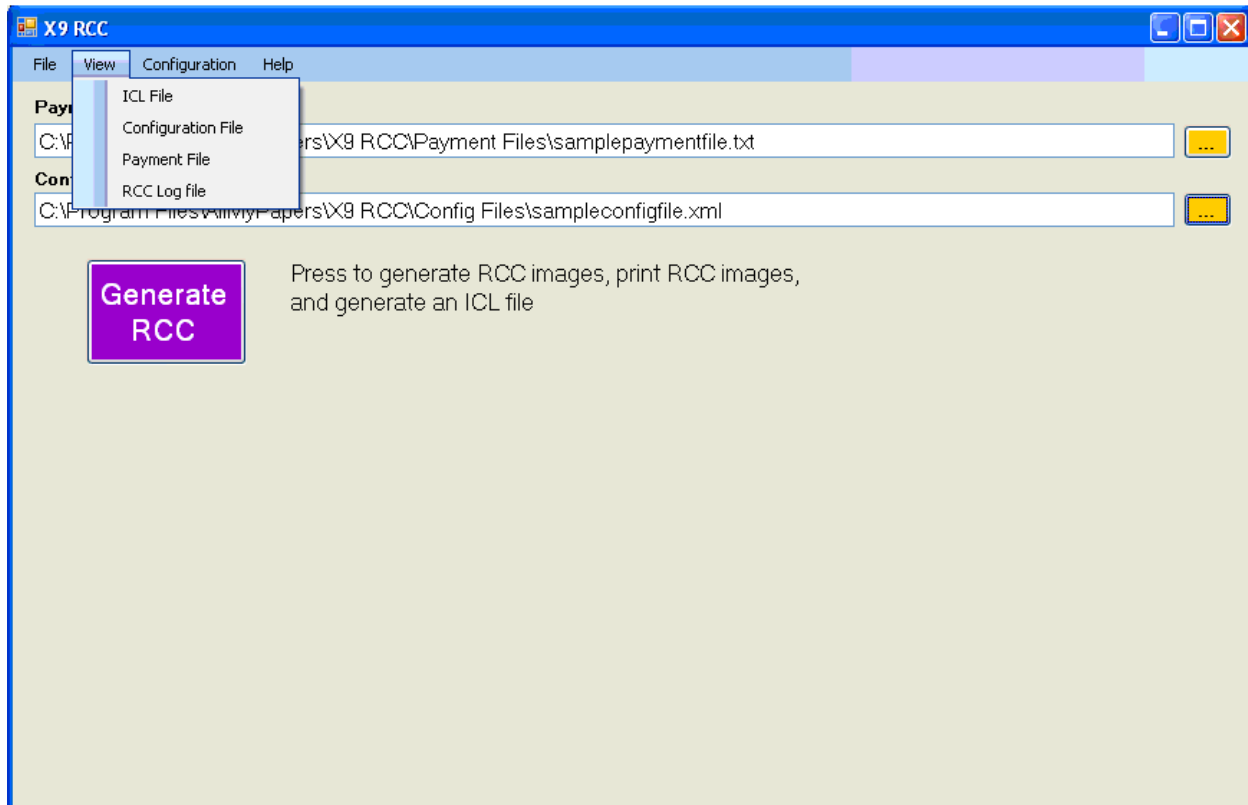


Figure 12 View Menu

8.3 Automatic Operations

The X9 RCC can be invoked as a command line function.

Command line syntax:

X9RCC <payment file path> <configuration file path>

For Merging Cash Letter Files

X9RCC -M <merge list pathname> <output ICL pathname>

Where the merge list file is a list of the ICL files to merge (1 per line)

It is recommended to delimit the arguments with double quotes to prevent problems with space characters in the pathnames.

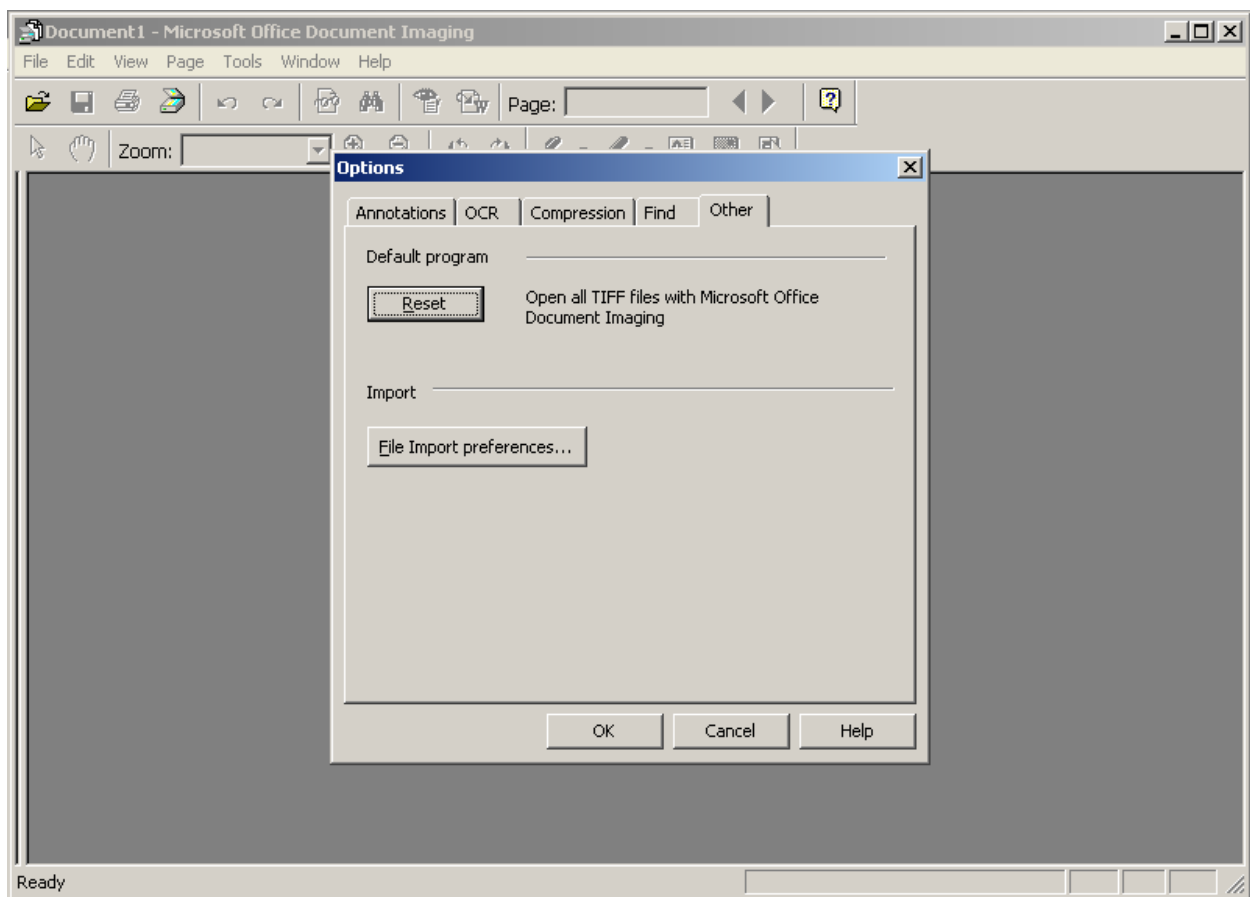
9 MICROSOFT DOCUMENT IMAGING SETUP

Microsoft Document Imaging can be used to TIFF image template files from MS Word document design. This application needs to be manually setup to save output as TIFF images.

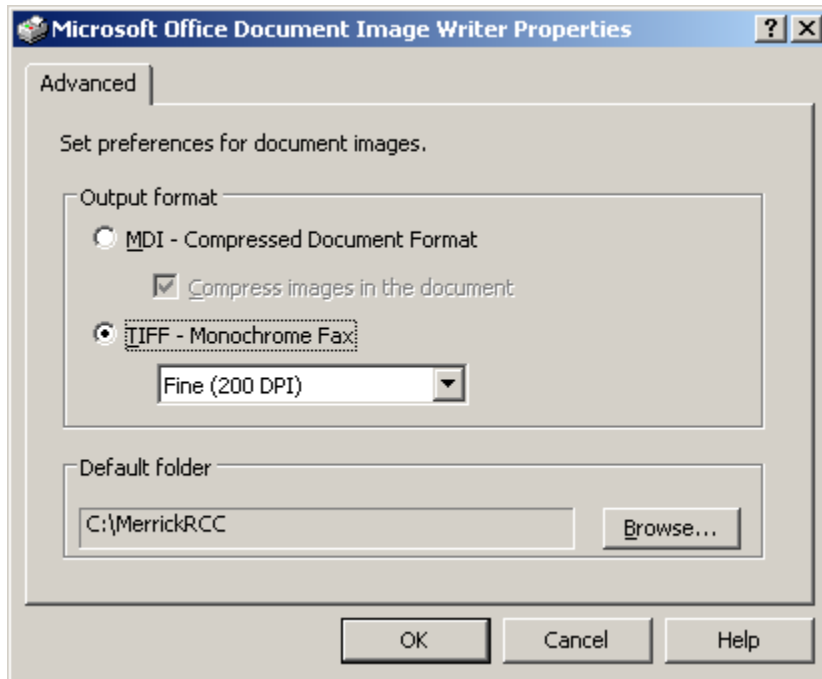
1. Locate and run "Microsoft Office Document Imaging" application. It should be located in C:\Program Files\Microsoft Office\Microsoft Office Tools



2. Select Menu item Tools -> Options, and select the "Other" tab.



3. Select File Import preferences.



4. Set TIFF – Monochrome Fax, Fine (200DPI)
5. Set “Default Folder” to the application directory you will be using for creating the RCC ICL files.
6. Click “OK”, “OK”, and close Microsoft Office Document Imaging.

APPENDIX A SAMPLE OF PAYMENT FILE



```
TEST CHECK FILE.txt - Notepad
File Edit Format View Help
"Franklin Templeton", "1234567890121654", "8888888888888888", "1.0000", "2/28/2008 12:00:00 AM", "9876", "2", "Abby Davis", "", "6548 Sterling
Circle", "", "Lunar", "WI", "56789", "Third Credit Union", "Lunar", "WI", "56789", "555444400", "770006332", "800-328-7448",
"GEICO Platinum MasterCard", "9876543210456709", "7777777777777777", "2.0000", "2/28/2008 12:00:00 AM", "5555", "2", "Mark Willis", "", "6844 Farrauto
Dr", "", "Hyde Mtn", "MA", "54321", "Seventh Street Bank", "Hyde Mtn", "MA", "45601", "5555111100", "988776650", "800-547-7277",
"Merrick Bank", "1234567891011121", "6666666666666666", "3.0000", "2/28/2008 12:00:00 AM", "1234", "1", "Tico Thomas", "Nstalle Anderson", "2718
Kennedy St", "Chen Park", "AZ", "00088", "Perrydog Bank", "Montoya", "AL", "15973", "3467106643", "666664440", "800-438-3253",
"New Millennium", "4444555566667777", "5555555555555555", "4.0000", "2/28/2008 12:00:00 AM", "6642", "1", "Lia Boca", "", "6621 Harper
Court", "Preston Island", "MT", "44400", "Raizer Federal Credit Union", "Preston Island", "MT", "66517
", "6718669047", "691866434", "800-392-5968"
```