



SOFTWARE DEFINED RADIO DEPARTMENT
SCA GROUP

**TUBITAK REFERENCE WAVEFORM
(TRWF) TUTORIAL**

Issue No	01	Project	TRWF
Issue Date	02.09.2009		
Issue Code		Language	English
Issue Type	Tutorial	Status	Final

Prepared by	Researcher	Adem ZUMBUL	
Approved by	Chief Researcher	Ibrahim OLCER	

UNCLASSIFIED

CHANGE LOG

<u>Version</u>	<u>Description</u>	<u>Date</u>
01	Initial release.	02.09.2009

The contents of this document are the property of TÜBİTAK
UEKAE and should not be reproduced, copied or disclosed to
a third party without the written consent of the proprietor.

© 2009 TÜBİTAK UEKAE
Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü
P.K. 74, Gebze, 41470 Kocaeli, TÜRKİYE
Tel: (0262) 648 1000, Faks: (0262) 648 1100

Bu dokümanın içeriği TÜBİTAK UEKAE'nin mülkiyetindedir.
Sahibinin yazılı izni olmadan çoğaltılamaz, kopyalanamaz ve
üçüncü şahıslara açıklanamaz.

Issue No: 01	Issue Date: 02.09.2009		2. of	10 Pages
--------------	------------------------	--	-------	----------

UNCLASSIFIED

CONTENTS

1. Introduction	4
1.1 Purpose	4
1.2 Abbreviations	4
2. Hardware Requirements	4
3. Software Requirements.....	4
4. Installation Steps for TRWF	5
5. Running TRWF	6

1. Introduction

This document provides an introduction to TUBITAK Reference Waveform (TRWF).

1.1 Purpose

The purpose of TRWF project is to provide a reference waveform implementation to support people trying to develop SCA compliant waveforms.

1.2 Abbreviations

CF – Core framework

SDR – Software Defined Radio

TRWF – TUBITAK Reference Waveform

WF – Waveform

2. Hardware Requirements

The following table presents the minimum and recommended hardware requirements to run TRWF.

<u>Hardware</u>	<u>Minimum</u>	<u>Recommended</u>
CPU Family	x86	x86
CPU Speed	1000 MHz	2000 MHz
RAM	512 MB	2 GB
Hard disk	5 GB	20 GB
Ethernet	100 Mbit	100 Mbit
Sound card	Full duplex sound card	Full duplex sound card

Table 1. Hardware Requirements

3. Software Requirements

The following table presents the software requirements to run TRWF.

<u>Software</u>	<u>Recommended</u>
OS	Linux (Pardus)
Core framework (CF)	SCARI-Open v2.2
CF ORB	Java ORB (1.4.12)
WF ORB	ACETAO (ACEv5.5, TAO1.5)
JAVA Compiler	JDK v1.5.0 build 16
C++ Compiler	GCC v3.4.6 and G++ v3.4.6.

Table 2. Software Requirements

4. Installation Steps for TRWF

This section lists the steps to install TRWF. TRWF can run on a single PC or two PCs connected over ethernet or WI-FI. The following steps have to be repeated on each PC.

1. Download and install Linux. TRWF is tested to run with Linux based Pardus 2008 operating system from TUBITAK. It can be downloaded from <http://www.pardus.org.tr/eng/index.html>.
2. Download and install required C++ and Java development packages. Pardus comes with a built in package manager which automates packet installation processes over internet connection. Development packages are autoconf, automake, binutils, ccache, cmake, gcc, gnuconfig, make, patch, pkgconfig, JDK. It is also recommended to install Eclipse IDE for C++ and Java development purposes.
3. TRWF is implemented with C++ and requires ACETAO as the CORBA layer. Download and install ACETAO from <http://www.cs.wustl.edu/~schmidt/TAO.html>. TRWF is tested to run with ACEv5.5 and TAOv1.5. Recommended installation path is “/ACE_wrappers”.
4. TRWF requires a SCA compliant CF. It has been tested to run with SCARI-Open which is an open source CF implementation from CRC. Download and install SCARI-Open v2.2 from http://www.crc.gc.ca/en/html/crc/home/research/satcom/rars/sdr/products/scari_open/scari2_downloads. Recommended installation path is “/home/Projects/java/SCARIv2.2/scari-Open”
5. Copy the “trwf” folder which comes with this tutorial (in folder “Deliverables/Distribution”) to “/home/trwf” directory and change its owner to a standart user. This folder includes TRWF related files.

<u>File</u>	<u>Description</u>
trwf.jar	TRWF installations file for SCARI-Open. It includes all executables, XMLs and DTDs.
config.txt	Configuration file for TRWF
input	Sample binary input file TRWF
input.txt	Sample text input file for TRWF

Table 3. Files in “/home/trwf” Folder

6. “/home/trwf/config.txt” keeps required settings for the TRWF. It looks as the following:

```
CONF_TCP_SERVER_IP
localhost
CONF_TCP_SERVER_PORT
2500
CONF_IS_SERVER
false
CONF_IS_LOOP_DATA
true
CONF_PACKET_SLEEP_TIME
10000
```

These settings are read by the TRWF on WF creation. The settings can be summarized as the following:

- CONF_TCP_SERVER_IP: This is the IP of the server waveform. Client TRWF reads this parameter to connect to the server TRWF. It is ignored on server PC and when the TRWF is running on a single PC.
- CONF_TCP_SERVER_PORT: This is the TCP port of the client and server. They must have the same port number for successful communication.

- CONF_IS_SERVER : This parameter determines the behavior of the TRWF. “true” means that TRWF will be server and “false” means that TRWF will be client. When running on a single PC this parameter is ignored. When running on two PCs, one of them should be client and the other should be server.
- CONF_IS_LOOP_DATA: If the TRWF will run on a single PC, this parameter should be set to “true” and if it will run on two connected PCs it could be set to “false” on both nodes.
- CONF_PACKET_SLEEP_TIME: This parameter sets the waiting time in micro second before sending each data packet to other PC. It should be adjusted according to the performance of the PCs. Default value is 10000. (Note that 1000000 means waiting 1 second between each packet)

5. Running TRWF

This section lists the steps to run TRWF.

1. Run naming service:

```
>> cd $SCA_HOME
>> ./startNamingService
```

```
umpc@umpc-pardus ~ $ cd $SCA_HOME
umpc@umpc-pardus scari-Open $ ./startNamingService

-----
Starting tnameserv on port 1050
-----

Initial Naming Context:
IOR: 00000000000000002b49444c3a6f6d672e6f72672f436f734e616d696e672f4e616d696e67436f6e74657
8744578743a312e30000000000001000000000000096000102000000000a3132372e302e302e3100041a00
000045afabcb0000000020000f4240000000010000000000000200000008526f6f74504f41000000000d5
44e616d65536572766963650000000000000080000000100000001140000000000020000000100000020
000000000001000100000002050100010001002000010109000000010001010000000026000000020002
TransientNameServer: setting port for initial object references to: 1050
Ready.
```

Figure 1. “./startNamingService” Output

2. Run SCARI-Open:

- a. In a new shell run DemoPlatformBootup script as the following:

```
>> cd $SCA_HOME
>> ./DemoPlatformBootup
```

```

sendDataToPort is not owned by root.
Since RFDevice will not be functional, it will not be started
*****

*****

sendDataToPort does not have UID set for root level execution.
Since RFDevice will not be functional, it will not be started
*****

-----
Node1.dcd.xml
-----

Node Started...

Press <Enter> to terminate nodeBoot

>

```

Figure 2. “./DemoPlatformBootup” Output

b. In a new shell run ApplicationManager script as the following:

```

>> cd $SCA_HOME
>> ./startApplicationManager

```

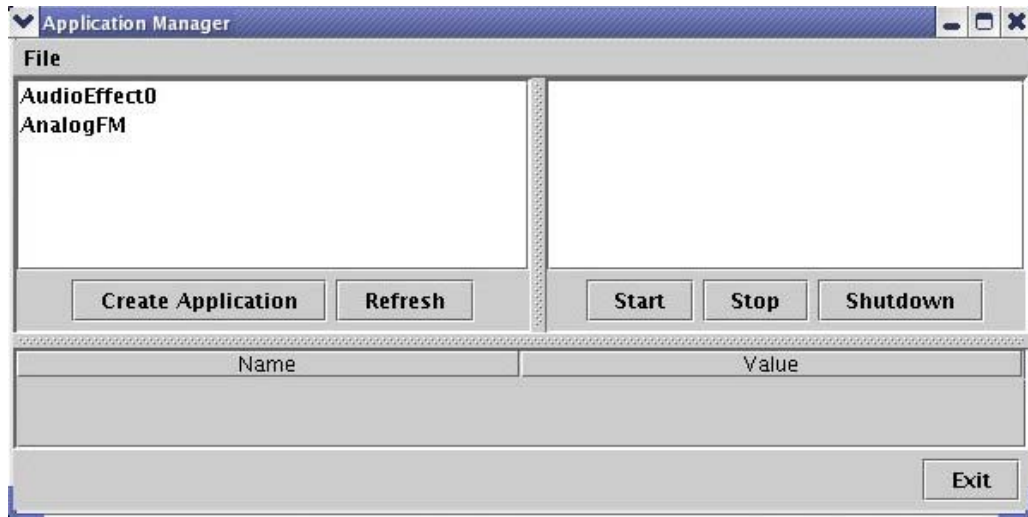


Figure 3. “./startApplicationManager” Output

c. Install TRWF to SCARI-Open

From File menu, select Install/Uninstall option to install and uninstall new waveforms to SCARI-Open. Install “/home/trwf/trwf.jar” file for installing TRWF to SCARI-Open.

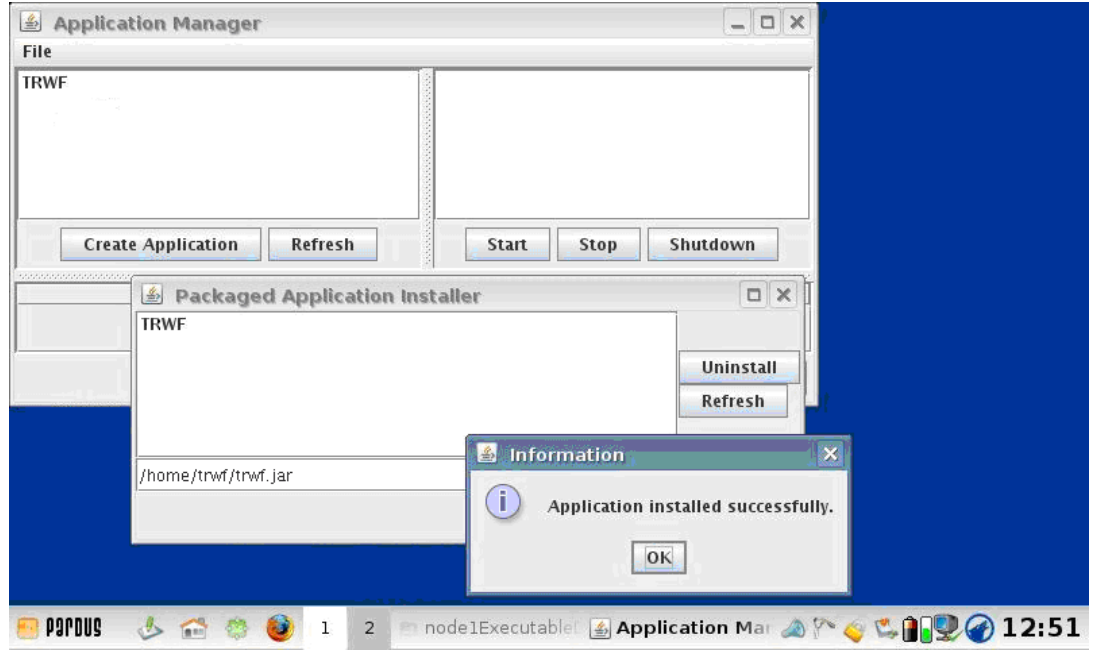


Figure 4. Installing TRWF

d. Create TRWF

After installing the trwf.jar to each node, TRWF application should be created by clicking “Create Application” button. A proper name should be written to the created WF when it is asked. It is recommended to use “client” and “server” names when running in two PCs in order to prevent naming confusions.

e. Configure TRWF

TRWF has several run-time properties that can be adjusted from CORBA by using ApplicationManager of the SCARI-Open. These settings are defined in XML files of the TRWF and they are loaded on waveform creation. These settings are read by the TRWF on creation. The settings can be summarized as the following:

- QoS: Stands for “Quality of Service”. This value is intended for future-use and can take values between 1 and 5. In this version this value is ignored but added to the header of data packets.
- DestinationId: This value is a 14 digit string such as (00905553332211) representing international GSM number of the destination node.
- Modulation: Number of bytes shifted. Transmitter shifts left, whereas receiver shifts right. This option shows that TRWF can process streaming data. For successful communication, modulation parameters should be same in each node. By playing with modulation parameters it can noticed that the data is received shifted by the difference of modulation parameters of the nodes. (Ex. If client has modulation parameter 10 and server has modulation parameter 9, this means the character ‘a’ will be received as ‘b’ because of the difference of modulation parameters). This parameter works for both voice mode and data mode.
- InputFilePath: When the application is in data transfer mode, this option represents input file path.
- ApplicationType: 0 for voice mode, 1 for data mode.

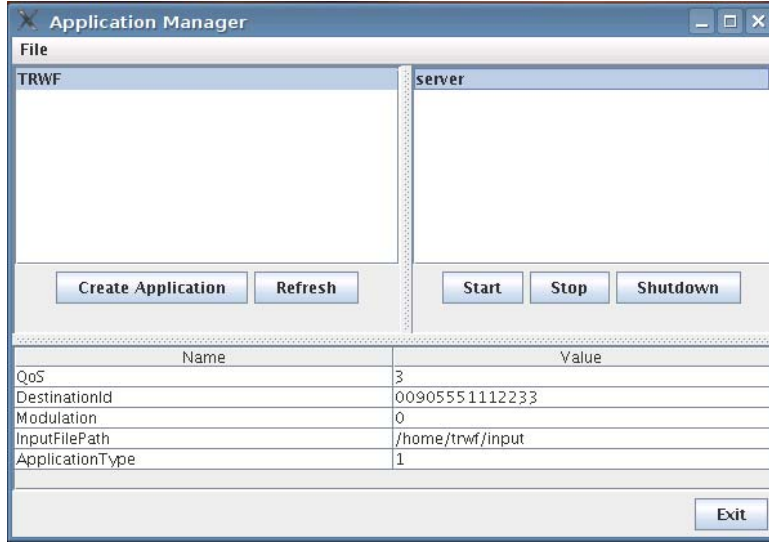


Figure 5. Configuration Parameters of TRWF

f. Start TRWF

TRWF can run on a single PC or two connected PCs.

- In order to run TRWF on a single PC, adjust CONF_IS_LOOP_DATA parameter to “true” in “/home/trwf/settings.txt” file and start TRWF from ApplicationManager. In this mode the input packets are looped by TRWF components. In voice mode (ApplicationType is 0), TRWF plays voice received from microphone and in data mode (ApplicationType is 1) TRWF reads file indicated by InputFilePath configuration and writes it to a file with a file name of the DestinationId parameter to the file system path of the ExecutableDevice of SCARI-Open which is located in “/home/Projects/java/SCARIV2.2/scari-Open/demosources/Node1/profile/node1ExecutableDevice_DCE_72f795a2-5f6a-41cf-8d3a-9beaeef0ac7d”.
- In order to run TRWF on two PCs, make sure that they have a valid network connection and they can ping each other successfully. Adjust “/home/trwf/settings.txt” in each node accordingly.

The following table shows example configuration files for client TRWF and server TRWF.

Server PC (10.0.0.1)	Client PC (10.0.0.2)
CONF_TCP_SERVER_IP 10.0.0.1	CONF_TCP_SERVER_IP 10.0.0.1
CONF_TCP_SERVER_PORT 2500	CONF_TCP_SERVER_PORT 2500
CONF_IS_SERVER true	CONF_IS_SERVER false
CONF_IS_LOOP_DATA false	CONF_IS_LOOP_DATA false
CONF_PACKET_SLEEP_TIME 10000	CONF_PACKET_SLEEP_TIME 10000

Table 4. Example Server and Client TRWF Configuration Files (“/home/trwf/settings.txt”)

After successfully creating and configuring applications on server and client, the TRWF can be started in both nodes. It is important that the server node should be started first, otherwise client cannot connect to server node. Once the connection is established in server and client, the order for stopping and restarting TRWF does not matter. When the server is started, it waits for client to be started. After the client is started communication begins in full-duplex mode. In other words, client and server can send and receive simultaneously. As in the single PC mode, outputs of TRWF is created in the file system path of SCARI-Open.