ERIKA Enterprise pre-built Virtual Machine

with support for Arduino, STM32, Xen Hypervisor and others

Version: 1.0

December 18, 2014



About Evidence S.r.l.

Evidence is a company operating in the field of software for embedded real-time systems. It started in 2002 as a spin-off company of the Real-Time Systems (ReTiS) Lab of the Scuola Superiore Sant'Anna (Pisa, Italy). Today, Evidence is a dynamic company having collaborations in the field of electronics, telecommunications, automotives, and industrial automation.

People at Evidence are experts in the domain of embedded and real-time systems, with a deep knowledge on the design and specification flow of embedded software, especially for the embedded market.

Besides providing consultancy services, Evidence also provides: BSPs based on Linux for embedded devices, evaluation boards featuring most innovative 8, 16 and 32-bit microcontrollers for the embedded market, development tools for making embedded software development easier, and tools for the schedulability analysis of real-time tasks running on your final product.

For more information see: http://www.evidence.eu.com

Contact Info

Evidence Srl, Via Carducci 56 Località Ghezzano 56010 S.Giuliano Terme PISA Italy

Tel: +39 050 99 11 224 Fax: +39 050 99 10 812

For more information about Evidence products, please send an e-mail to the following address: info@evidence.eu.com. Other information about the Evidence product line can be found at the Evidence web site at: http://www.evidence.eu.com.

This document is Copyright 2011-2012 Evidence S.r.l.

Information and images contained within this document are copyright and the property of Evidence S.r.l. All trademarks are hereby acknowledged to be the properties of their respective owners. The information, text and graphics contained in this document are provided for information purposes only by Evidence S.r.l. Evidence S.r.l. does not warrant the accuracy, or completeness of the information, text, and other items contained in this document. Matlab, Simulink, Mathworks are registered trademarks of Matworks Inc. Microsoft, Windows are registered trademarks of Microsoft Inc. Java is a registered trademark of Sun Microsystems. in the USA. and other countries, and are used under license. All other trademarks used are properties of their respective owners. This document has been written using $IAT_{\rm E}X$.

Contents

1	Intr	oductio	n	6
	1.1	Requir	rements	6
	1.2	Suppor	rted boards	6
	1.3	Licens	ing	7
	1.4	Feedba	ack, bugs, and additional examples	7
2	Inst	alling tl	he VM and creating the Virtual Machine settings	8
	2.1	Virtua	lBox	8
		2.1.1	Installing VirtualBox	8
		2.1.2	Downloading the Virtual Machine	8
		2.1.3	VirtualBox settings	9
	2.2	VMwa	re	9
		2.2.1	Installing VMware Player	9
		2.2.2	Downloading the Virtual Machine	10
		2.2.3	Creating the Virtual Machine on VMware	10
	2.3	Startin	ng the virtual machine	13
	2.4	Install	ing the VMware tools	16
3	Con	npiling t	the ERIKA Enterprise demo application	18
4	Pro	grammi	ng the boards	21
5	ERI	KA Ent	erprise as a Xen domU	24
6	Ack	nowled	gments	25

List of Figures

2.1	Files obtained when unpacking the virtual machine. Actual file names
	may vary
2.2	VirtualBox opened after clicking on the vbox file (the one with the blue
	icon)
2.3	Opening VMware 10
2.4	Create a new VMware virtual machine
2.5	Select that you do not want to install an operating system
2.6	Select linux/Ubuntu as Guest OS
2.7	Select to Store the virtual disk in a single file
2.8	Provide a name for the virtual machine
2.9	You just created the virtual machine
2.10	VMware settings. We suggest 1 GB of system memory
2.11	Login screen of the Virtual machine
2.12	Welcome page
2.13	Open a terminal
2.14	Commands to give to install the VMware tools
3.1	Eclipse requires the workspace location. Leave the default setting 18
3.2	The Eclipse welcome screen
3.3	Provide a name for the Eclipse project containing the demo 19
3.4	How to select an Arduino Uno / Arduino Nano example
3.5	How to select an STM32F4Discovery example
4.1	How to connect the STM32F4Discovery to the Virtual Machine 22
4.2	Flashing an ERIKA executable on STM32F4Discovery
4.3	Flashing an ERIKA executable on Arduino Nano

About this document

This document describes the installation, first setup and first demo run of the ERIKA Enterprise pre-built Virtual Machine for Arduino, STM32F4Discovery, and other boards. This Virtual Machine was done in collaboration with the ReTiS Lab of the Scuola Superiore S. Anna, and the University of Modena Reggio Emilia.

Function of the document

The function of this document is to provide a quick start guide for using the Virtual Machine with a demo example, plus a link to all information needed to start ERIKA Enterprise under a Xen domU.

Document history

Version	Date	Author	Company	Change Description
1.0	July 2014	Paolo Gai	Evidence Srl	Initial version
1.1	December 2014	Paolo Gai	Evidence Srl	Added XEN support

1 Introduction

Installing a complete development and debugging environment for an embedded board always involves a lot of work in installing compilers, debuggers, development environments, makefiles, and so on. This Virtual Machine is aimed at providing a quick solution for all these problems, providing a Linux platform with all software preinstalled and ready to work, allowing you to compile and flash OSEK/VDX applications on selected target boards.

In other words, this Virtual Machine provides a complete virtual environment where you will be able to:

- Edit your OSEK/VDX application using the open-source OSEK/VDX Kernel ERIKA Enterprise;
- Compile your application using the pre-installed open-source compilers;
- Program target boards connected to the PC using an USB port routed to the Virtual Machine.

In addition to this, this Virtual Machine contains a pre-built version of ERIKA Enterprise which works as a domU on the Xen Hypervisor (see Chapter 5).

1.1 Requirements

We provide two versions of this Virtual Machine. One for VirtualBox (https://www.virtualbox.org/), and one for VMware Player (https://my.VMware.com/web/VMware/free).

On Linux Hosts, both work fine. On Windows hosts, we noticed communication problems on VirtualBox when connecting a STM32F4Discovery¹. For this reason, if you are using an STM32F4Discovery on a Windows host we suggest to use VMware Player, which is free for non commercial usage.

1.2 Supported boards

We tried the virtual machine described in this document with the following boards:

- STM32F4Discovery;
- Arduino Uno;

¹See https://forums.virtualbox.org/viewtopic.php?f=6&t=54026

- Arduino Nano;
- Cubieboard2 (for the Xen domU integration, see Chapter 5).

1.3 Licensing

The Virtual Machine described in this document includes various open-source software. The following points shortly describe the main licenses of the tools which has been integrated during this work:

- The Linux Distribution is a standard Ubuntu distribution. For more information about Ubuntu and the software licenses included in this Linux distribution please refer to the following website: http://www.ubuntu.com/.
- ERIKA Enterprise is distributed mainly under the GPL2+Linking Exception License (http://en.wikipedia.org/wiki/GPL_linking_exception), plus other licenses used for user libraries in contrib directory.
- Eclipse, EMF, and the RT-Druid plugins are distributed under the EPL License (http://en.wikipedia.org/wiki/Eclipse_Public_License).
- The additional compiler for AVR and Cortex-M pre-installed on this virtual machine are based on GCC, which is distributed under the GNU GPL License.
- Xen, linux, and other software used on the Cubieboard2 are distributed under the GPL License.

1.4 Feedback, bugs, and additional examples

We care about your feedback! Information, feedback, and new demos about ERIKA Enterprise can be provided directly on the ERIKA Enterprise website:

http://erika.tuxfamily.org

For commercial technical support, sales, pricing, order status, and general information and feedback, please contact Evidence Srl directly at the address and phone numbers available at the following web page:

http://www.evidence.eu.com/en/contact-us.html

2 Installing the VM and creating the Virtual Machine settings

2.1 VirtualBox

2.1.1 Installing VirtualBox

VirtualBox can be freely downloaded and used also for commercial use from the following website:

https://www.virtualbox.org/wiki/Downloads

All you need to do is to download the VirtualBox installer, and install it on your PC. All the following screenshots will refer to the usage of VirtualBox on a Windows 7 Host machine.

2.1.2 Downloading the Virtual Machine

The Virtual Machine can be downloaded from the following website:

http://www.erika-enterprise.com

The Virtual Machine is typically distributed as a compressed file. Please decompress it. You will find at least two files, as in Figure 2.1. The file with the vbox extension is the file containing the settings of the virtual machine (describing the guest hardware, memory, disks, ...). The file with the vdi extension is the virtual hard disk used by the virtual machine.

On a typical VirtualBox setup, just double clicking on the file with vbox extension will open VirtualBox as in Figure 2.2. Just click on the Start button to boot the Virtual Machine.

G v 🐌 → Com	puter	> System (C:) > pj > vm > virtualbox_erika_diab_lauterbach_ppc_marr	iba		• 4 ₇	Cerca virtu 🔎
Organizza 👻 Inclu	ıdi ne	la raccolta 🔻 Condividi con 👻 Nuova cartella			8==	- 🗌 🔞
☆ Preferiti B Download ♥ Dropbox Risorse recenti ■ Desktop Gogle Drive	4 III +	Nome 2020 Very M. Machine with addition update 28,05,2014 RTD EL VM_Machine with addition update 28,05,2014 RTD-disk1	Ultima modifica 12/06/2014 12:05 12/06/2014 10:36	Tipo VirtualBox Machin Virtual Disk Image	Dimensione 10 KB 7.629.892 KB	
2 element	i					

Figure 2.1: Files obtained when unpacking the virtual machine. Actual file names may vary.

Oracle VM VirtualBox Manager File Machine Help	
New Settings Start Discard	🚱 Details 💿 Snapshots
New Settings Start Discard	General Name: E, VM, Machine with addition update 28, 05, 2014 RTD Operating System: Ubuntu (32 bit) Image: System: Ubunu (32 bit) Image: System: <
	Host Driver: Windows DirectSound

Figure 2.2: VirtualBox opened after clicking on the vbox file (the one with the blue icon).

2.1.3 VirtualBox settings

The following is a list of the main settings of the Virtual machine, useful if, for some reason, you need to recreate the **vbox** file from scratch. Those information must be set on a new virtual machine by clicking on the "Settings" button in Figure 2.2.

- 1. General Tab, Basic subtab: The Type of virtual machine must be Linux / Ubuntu 32 bit.
- 2. System tab: We suggest a 1Gb system memory, I/O APIC active, and as many processors as you have in your physical machine.

Please note that the virtual machine comes with the VirtualBox Guest Additions already installed. This turns out to be very convenient as the X Server will automatically recognize a resize of the VirtualBox window.

2.2 VMware

2.2.1 Installing VMware Player

VMware Player can be freely downloaded and used (not for commercial use) from the following website:

https://my.VMware.com/web/VMware/free



Figure 2.3: Opening VMware.

All you need to do is to download the VMware, and install it on your PC. All the following screenshots will refer to the usage of VirtualBox on a Windows 7 Host machine.

2.2.2 Downloading the Virtual Machine

The Virtual Machine can be downloaded from the following website:

```
http://www.erika-enterprise.com
```

The Virtual Machine is typically distributed as a compressed file. Please decompress it. You will find one file, which is the virtual hard disk used by the virtual machine.

2.2.3 Creating the Virtual Machine on VMware

To open the virtual Machine, please follow the steps below:

- 1. Start VMware Player as shown in Figure 2.3.
- 2. Create a new Virtual Machine as shown in Figure 2.4.
- 3. Select that you will not install the OS, as in Figure 2.5.
- 4. Select Linux / Ubuntu as the operating system, as in Figure 2.6.
- 5. Select that you want to store the VM hard disk in a single file as in Figure 2.7.

-	VMware Player (Non-commercial use only)				
PI	ayer 🗸 📄 🔹 🗐 🍯					
	File	•	*	New Virtual Mac	hine Ctrl+N	
C) Power	Þ		Open	Ctrl+0	re Player
	Removable Devices	Þ		Download Virtua	I Appliance	
	Send Ctrl+Alt+Del			Player Preference	es	al Machine
2	Full Screen Ctrl+Alt+Ent	er	-	T T	added to the top of your libra	, which will then be ry.
Ę	h Unity					
1	3 Manage	Þ			Onen a Virtual Mac	hine
	Help	►		5	Open an existing virtual machine, which will then	
	Exit				added to the top of your libra	ry.
Т						
					Upgrade to VMware	e Workstation
					Get advanced features such a developer tool integration, an	as snapshots, id more.
					Help	
					View VMware Player's help co	ntents.
					This product is not licensed ar non-commercial use only. For	nd is authorized for commercial use,
					purchase a license. Buy now.	

Figure 2.4: Create a new VMware virtual machine.

New Virtual Machine Wizard	×
Welcome to the New Virtual Machine Wizard A virtual machine is like a physical computer; it needs an operating system. How will you install the guest operating system?	
Install from:	
🔘 Installer disc:	
No drives available 👻	
⊙ Installer disc image file (iso):	
Brows	e
I will install the operating system later.	
The virtual machine will be created with a blank hard disk.	
Help < Back Next > Ca	ancel

Figure 2.5: Select that you do not want to install an operating system.

New Virtual Machine Wizard	X
Select a Guest Operating System Which operating system will be installed on this virtual machine?	
Guest operating system	
O Microsoft Windows	
Linux	
🔘 Novell NetWare	
Solaris	
🔘 Other	
Version	
Ubuntu	-
Help < Back Next > Ca	ancel

Figure 2.6: Select linux/Ubuntu as Guest OS.

New Virtual Machine Wizard	8
Specify Disk Capacity How large do you want this disk to be?	
The virtual machine's hard disk is stored as one or more files on the host computer's physical disk. These file(s) start small and become larger as you add applications, files, and data to your virtual machine.	
Maximum disk size (GB): 20.0	
Recommended size for Ubuntu: 20 GB	
Store virtual disk as a single file	
Split virtual disk into multiple files	
Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.	
Help <a>Back Next > Can	cel

Figure 2.7: Select to Store the virtual disk in a single file.

New Virtual Machine Wizard	—
Name the Virtual Machine What name would you like to use for this virtual machine?	
Virtual machine name:	
ERIKA Enterprise for STM32FDiscovery and Arduino Uno	
Location:	
C:\pj\vm\vmware_RETIS_STM32_Arduino	Browse
< Back Next >	Cancel

Figure 2.8: Provide a name for the virtual machine.

6. Provide a name for the virtual machine, as in Figure 2.8.

As a result, you will get a dialog box like in Figure 2.9. You can edit the virtual machine settings as you wish before saving (see Figure 2.10). We suggest to allocate 1Gb of memory.

At this point, go into the directory where you created the virtual machine. There will be a file with extension .vmdk. All you need to do is to replace that file with the one found in the compressed archive (note the file name should be maintained as the one created by VMware).

2.3 Starting the virtual machine

Once powered up, Linux will be boot, and the login screen of the Ubuntu Distribution will appear as in Figure 2.11.

To log in, use the following credentials:

Username: evidence

Password: evidence

Once logged in, you will get an HTML welcome page as in Figure 2.12. The welcome page contains a set of documents on the following topics:

• ReTiS Lab information;

New Virtual Machine Wizard	— —
Ready to Create Virtual Machine Click Finish to create the virtual machine. Then you can install Ubunt	ນ.
The virtual machine will be created with the following settings:	
Name: ERIKA Enterprise for STM32FDiscovery and Arduino	🔺
Location: C:\pj\vm\vmware_RETIS_STM32_Arduino	
Version: Workstation 10.0	
Operating System: Ubuntu	Ш
Hard Disk: 20 GB	
Memory: 1024 MB	
Network Adapter: NAT	
Other Devices: CD/DVD, USB Controller, Printer, Sound Card	-
Customize Hardware	
< Back Finish C	ancel

Figure 2.9: You just created the virtual machine.

levice	Summary 1 GB 1 Auto detect NAT Present Auto detect Present Auto detect	Specify the ai machine. The Memory for th 44 GB - 32 GB - 16 GB - 8 GB - 4 GB - 2 GB - 1 GB - 512 MB - 256 MB - 128 MB - 128 MB - 32 MB - 16 MB - 8 MB - 4 MB -	nount of memory allocated to this virtual memory size must be a multiple of 4 MB. is virtual machine: 1024 m MB Maximum recommended memory (Memory swapping may occur beyond this size.) 2852 MB Recommended memory 1024 MB Guest OS recommended minimum 512 MB
--------	--	--	---

Figure 2.10: VMware settings. We suggest 1 GB of system memory.



Figure 2.11: Login screen of the Virtual machine.



Figure 2.12: Welcome page.

- Evidence Professional support on ERIKA Enterprise.
- Tutorial documentation (this file);

Finally, remember you can always click on the zoom icon to go full screen.

2.4 Installing the VMware tools

The VMware machine comes without VMware tools installed. To install them, select the installation of the VMware tools from the Player menu. Afterwards, open a terminal (see Figure 2.13) and follow the instructions in Figure 2.14. You can add the option -d to provide the default answers to all questions. Once installed, reboot the virtual machine, and once logged the screen size will adapt to the size of the virtual machine window.



Figure 2.13: Open a terminal.

📲 ERIKA Enterprise for STM32FDiscovery and Arduino Uno - VMware Player (Non-commercial use only)			
Player 👻 🔡 👻 📇 📜	C 🐻		« 📄 🖶
😢 🖻 🖻 File Edit Vie	w Search Terminal Help	🔀 🖇 👣 🕪)) 2:3:	3 PM 👤 evidence 🔱
evidence@evi evidence@evi 9.6.2-16883 evidence@evi 9.6.2-16883 evidence@evi evidence@evi bin doc et evidence@evi bin doc et evidence@ev	dence-VirtualBox:-\$ mkdir vr dence-VirtualBox:-\$ cd vntor dence-VirtualBox:-/vntools\$ 56.tar.gz dence-VirtualBox:-/vntools\$ idence-VirtualBox:-/vntools\$ idence-VirtualBox:-/vntools/ word for evidence: ce-VirtualBox:/home/evidence/	ntools Js/ tar xzf /media/VMware\ T ls cd vmware-tools-distrib/ r lib vmware-install.pl /mware-tools-distrib\$ sud /vmtools/vmware-tools-dis	ools/VMwareTools- o su trib# ./vmware-in
Click in the virtual screen keystrokes	Take sure that you are logged in to the guest operati aunch a Terminal, and use tar to uncompress the inst Tools.	ng system. Mount the virtual CD drive in the g aller. Then, execute vmware-install.pl to instal	uest, X William Help

Figure 2.14: Commands to give to install the VMware tools.

3 Compiling the ERIKA Enterprise demo application

The following steps will guide you in the compilation of a simple ERIKA Enterprise application for Arduino Uno and for STM32F4Discovery Kit:

- 1. To compile your first application with ERIKA Enterprise, you need to open the Eclipse IDE. There is an Eclipse RT-Druid link on the Desktop.
- 2. Double click on it, and Eclipse will open requiring the workspace location.Please confirm the default location /home/evidence/workspace as in Figure 3.1. The Eclipse welcome screen will appear as in Figure 3.2. Click on the Workbench icon, and the default Eclipse view will appear.



Figure 3.1: Eclipse requires the workspace location. Leave the default setting.

- 3. Click on the "New" button in the toolbar (the first on the left), and choose "RT-Druid Oil and C/C++ project". A Dialog Box will appear as in Figure 3.3. Provide a name for the project, and press Next.
- 4. Select the checkbox "Create a project using one of these templates", and select either a demo for the Arduino Uno board (see Figure 3.4) or for the STM32F4Discovery board (see Figure 3.5).
- 5. Click on the Finish button to create the example.



Figure 3.2: The Eclipse welcome screen.

C/C++
°
ailable.
7

Figure 3.3: Provide a name for the Eclipse project containing the demo.

- 6. At this point, you can explore the demo examples:
 - They are typically composed by two files, a conf.oil file, containing the OIL description needed to statically configure the kernel, and code.c, containing the application code.
 - The target board selected for the specific demo is listed in the OIL file parameters. Note that for Arduino also Arduino Nano is supported, but the specific option have to be uncommented in the OIL file.
- 7. To compile the project, just right click on the project name and select "Build Project". As a result, the project is compiled using the GCC cross-compiler for the specific board. The output is printed on the Console view.

You are now ready to program the resulting files on the target board.



Figure 3.4: How to select an Arduino Uno / Arduino Nano example.



Figure 3.5: How to select an STM32F4Discovery example.

4 Programming the boards

To program the Arduino Uno, Arduino Nano and STM32F4Discovery from the virtual machine, you can do the following steps:

- 1. Connect the board to the PC. A driver will be installed for the board in the Windows host.
- 2. You now need to "connect" the board to the Virtual Machine. This connection is a redirection of all USB packets from the Windows host to the Linux Guest OS hosted in the Virtual Machine. Figure 4.1 shows how to do that on VMware for the STM32F4Discovery board.
- 3. Open a terminal, and change the directory to the **Debug** directory created when compiling the project.
- 4. After that, you can run the script for flashing the compiled file into the board.
 - For the STM32F4Discovery Kit board, you can use the command:

stmflash

which will flash the default binary created by ERIKA inside the board (see Figure 4.2).

• For the Arduino Uno board, you can use the command:

arduinounoflash

or, as an alternative, try:

arduinoflash -b 115200

• For the Arduino Nano board, you can use the command:

arduinonanoflash

or, as an alternative, try:

```
arduinoflash -d /dev/ttyUSB0
```

as shown in Figure 4.3.

The scripts described above basically search the output file of the ERIKA compilation in a set of default directories, and afterwards apply some default parameters which *typically* work. In case the default parameters do not work, additional parameters can be specified. In particular, the **arduinoflash** script accepts two parameters:



Figure 4.1: How to connect the STM32F4Discovery to the Virtual Machine.



Figure 4.2: Flashing an ERIKA executable on STM32F4Discovery.



Figure 4.3: Flashing an ERIKA executable on Arduino Nano.

- -d device can be used to specify the device where the Arduino board is connected. The default is typically good for Arduino Uno, and is: /dev/ttyACM0
- -b baudrate can be used to specify the baud rate to be used for the serial line. The default is typically good for Arduino Nano and is 57600.

The stmflash script accepts various parameters that can be obtained by calling the script with the -h option.

5 ERIKA Enterprise as a Xen domU

Thanks to the work of Arianna Avanzini, with the supervision of Prof. Paolo Valente from the University of Modena Reggio Emilia¹, ERIKA is now able to be run as a Xen domU, in an integrated Multi-OS system with Embedded Linux on one core, and ERIKA Enterprise on the other core.

The porting currently supports the Cubieboard 2^2 . The Virtual Machine contains all the software, compilers, and environment needed to compile and run (on the target board) a complete Multi-OS system.

For more information on how to proceed, please refer to the following wiki page:

http://erika.tuxfamily.org/wiki/index.php?title=Xen_Hypervisor

¹http://algogroup.unimore.it/people/paolo/ ²http://cubieboard.org/model/cb2/

6 Acknowledgments

We would like to acknowledge Mauro Marinoni, Gianluca Franchino, Alessandro Biondi, Pasquale Buonocunto, Pietro Lorefice, and Daniel Cesarini from the ReTiS Lab, and Bruno Morelli, Francesco Esposito, Giuseppe Serano from Evidence for the support provided during the development of the demos, and during integration phase of this Virtual Machine.

We also thank Paolo Valente and Arianna Avanzini for the Xen integration.