



in 15 minutes-

User Guide

Rev. 1.3

Date : 14th of January, 2014

oStudio - Live Tuning's Getting Started

www.objectis-software.com www.objectis-software.com/support software@objectis.ch



Table of contents

1	Intr	oduction	3
2	Step	o by step procedure	3
	2.1	Download and setup	3
	2.2	Copy Ocfe files to Arduino IDE libraries	3
	2.3	Extract the example	4
	2.4	Compile and upload the example to your Arduino target	5
	2.5	Test and enjoy with oStudio LiveTuning	7
3	Mo	re to do with oStudio and LiveTuning	.14



Live Tuning for Arduino with OcfEmbedded

1 Introduction

Arduino, which received an Honorary Mention in the Digital Communities section of the 2006 Ars Electronica Prix, is now totally comptaible with Ocf Server. It means that every client (can be oStudio – Live Tuning or your own application with Ocf Client) can connect to an Arduino single-board microcontroller. As usual, the client can interact with the Arduino board LIVE!



2 Step by step procedure

2.1 Download and setup

- Download and install oStudioLive Tuning from Objectis web site www.objectis-software.com
- Download and open the ArduinoOcfe_Application_Note.zip

2.2 Copy Ocfe files to Arduino IDE libraries

From the ArduinoOcfe_Application_Note.zip file, folder "libraries" :

	rduinoOcfr	a_Applicatio	n_Note_1	_0.zip\Lik	oraries\						
<u>F</u> ichier	Edition	Affic <u>h</u> age	Fa <u>v</u> oris	<u>O</u> utils	<u>A</u> ide						
- e	1	_	\checkmark	er	⇒	-	×	ñ			
Ajout	Ajouter Extraire Tester Copier Déplacer Supprimer Informations										
ø 🛛	D:\Ardui	inoOcfe_Ap	plication_1	Note_1_0	.zip\Librari	ies\					
Nom					Tai	lle	Compressé	Modifié le	Créé le	Accédé le	Attributs
Cofe	4				24 50	01	5 566	2014-01-03 12:05	2014-01-03 12:05	2014-01-03 12:05	D
<				III							
1 objet(s) sélection	né(s) 24	4 501	2	4 501	20)14-01-03 12:0	05			



extract the folder "Ocfe" to the "Arduino IDE" libraries folder (which is typically <u>C:\Program Files</u> (x86)\Arduino\libraries) :

🕞 🕞 🗢 📙 « Program Files (x80	6) ▶ Arduino ▶ libraries ▶	👻 🍫 Rechercher dans :	libraries 🔎
Organiser 🔻 湇 Ouvrir 🛛 Ir	nclure dans la bibliothèque 🔻	Partager avec 🔻 Graver 🚿	:= • 🔟 🔞
🚖 Favoris	Nom	▼ Modifié le	Туре Та
	EEPROM	02.01.2014 11:13	Dossier de fichiers
🥽 Bibliothèques	퉬 Esplora	02.01.2014 11:13	Dossier de fichiers
Documents	퉬 Ethernet	02.01.2014 11:13	Dossier de fichiers
📔 Images	퉬 Firmata	02.01.2014 11:13	Dossier de fichiers
🎝 Musique	퉬 GSM	02.01.2014 11:13	Dossier de fichiers
Subversion	퉬 LiquidCrystal	02.01.2014 11:13	Dossier de fichiers
📑 Vidéos	🌗 Ocfe	02.01.2014 11:17	Dossier de fichiers
	Robot_Control	02.01.2014 11:13	Dossier de fichiers
🖳 Ordinateur	Robot_Motor	02.01.2014 11:13	Dossier de fichiers
🚢 Disque local (C:)	🔰 SD	02.01.2014 11:13	Dossier de fichiers
🧫 Données (D:)	퉬 Servo	02.01.2014 11:13	Dossier de fichiers
🧳 LaCie (G:)	퉬 SoftwareSerial	02.01.2014 11:13	Dossier de fichiers
	퉬 SPI	02.01.2014 11:13	Dossier de fichiers
🗣 Réseau	퉬 Stepper	02.01.2014 11:13	Dossier de fichiers
	퉬 TFT	02.01.2014 11:13	Dossier de fichiers
	퉬 WiFi	02.01.2014 11:13	Dossier de fichiers
	퉬 Wire	02.01.2014 11:13	Dossier de fichiers
			+
Ocfe Mod Dossier de fichiers	ifié le : 02.01.2014 11:17		

2.3 Extract the example

From the ArduinoOcfe_Application_Note.zip file, folder "Example":

Z D:\Arc	D:\ArduinoOcfe_Application_Note_1_0.zip\ArduinoOcfe_Application_Note_1_0\Example\								
<u>Fichier</u>	Edition Affichag	ge Fa <u>v</u> oris	<u>O</u> utils <u>A</u> ide						
- •		\checkmark	•		×	ĩ			
Ajoute	r Extraire	Tester	Copier	Déplacer	Supprime	r Informations			
ø 🐌	D:\ArduinoOcfe_	Application_N	Note_1_0.zip\Ar	duinoOcfe_A	pplication_No	te_1_0\Example\			•
Nom				Taille	Compressé	Modifié le	Créé le	Accédé le	Attributs
📗 Ardui	noOcfe			3 203	1 012	2014-01-03 12:15	2014-01-03 12:08	2014-01-03 12:15	D
•	< III							Þ	
1 objet(s)	sélectionné(s)	3 203	3 203	20	014-01-03 12:1	5			

Extract the ArduinoOcfe folder to the folder where you usually store your Arduino projects.



Live Tuning for Arduino with OcfEmbedded

2.4 Compile and upload the example to your Arduino target

Open the ArduinoOcfe.ino file with "Arduino Ide"



Select the Board model matching your Arduino

💿 ArduinoOcfe Ardu	ArduinoOcfe Arduino 1.0.5					
File Edit Sketch Too						
ArduinoOcfe	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload		<u>م</u> •			
/* ArduinoOcfe	Serial Monitor Ctrl+Shift+M		<u>^</u>			
	Board •	Arduino Uno				
Demonstrate:	Serial Port +	Arduino Duemilanove w/ ATmega328	=			
This example	Programmer 🕨	Arduino Diecimila or Duemilanove w/ ATmega168 Arduino Nano w/ ATmega328 Arduino Nano w/ ATmega328				
Pre-requisites	:	Arduino Nano W/ Armegatos				
- the OCFE fold	der containing protocol files mu	Arduino Maga (ATmaga 1290)				
TIDIALIES IO.	ider (cypically c.,riogram file	Arduino Leonardo				
More information at : www.objectis-software.co		Arduino Esplora				
*/		Arduino Micro				
#include "OcfEmb	edded.h"	Arduino Mini w/ ATmega328				
		Arduino Mini w/ ATmega168				
<pre>int AnalogIn[8]; int DigitalIn[14</pre>	1:	Arduino Ethernet				
int DigitalOut[1	4];	Arduino Fio				
int BlinkPeriod[.	14];	Arduino BT w/ ATmega328				
OCFE BEGIN(ocfEn	tries)	Arduino BT w/ ATmega168				
OCFE_VARIABLE_	READONLY(AnalogIn[0], OCFE_TYPE_	LilyPad Arduino USB				
OCFE_VARIABLE_	READONLY(AnalogIn[1], OCFE_TYPE_	LilyPad Arduino w/ ATmega328				
OCFE_VARIABLE_I	READONLY(AnalogIn[2], OCFE_TYPE] READONLY(AnalogIn[3], OCFE_TYPE	LilyPad Arduino w/ ATmega168				
OCFE_VARIABLE_	READONLY(AnalogIn[4], OCFE_TYPE	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328				
OCFE_VARIABLE_	READONLY(AnalogIn[5], OCFE_TYPE_	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168				
OCFE_VARIABLE_	READONLY(AnalogIn[6], OCFE_TYPE_	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328				
		Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168				
		Arduino NG or older w/ ATmega168				
		Arduino NG or older w/ ATmega8				
		Arduino Robot Control				
		Arduino Robot Motor				
1		Arduino M	lega 2560 or Mega ADK on COM15			



Connect your Arduino to USB port and select the corresponding communication port:

💿 ArduinoOcfe Ardu	ino 1.0.5			
File Edit Sketch Too	ls Help			
ArduinoOcfe	Auto Format Archive Sketch Fix Encoding & Reload	Ctrl+T		ø. ▼
/* ArduinoOcfe	Serial Monitor	Ctrl+Shift+M		×
Demonstrate	Board Serial Port	• •	COM15 protocol on Arduino.	
This example	Programmer Burn Bootloader	•		
1			Arduino Meg	a 2560 or Mega ADK on COM15

Compile and upload to the Arduino

💿 ArduinoOcfe Arduino 1.0.5	
File Edit Sketch Tools Help	
	P
ArduinoOcfe	
/* ArduinoOcfe	A
Demonstrates the capabilities and ease of use of the Ocfe protocol on Arduino.	_
This example code is in the public domain.	F
Pre-requisites : - the OCFE folder containing protocol files must be copied into the Arduino IDE libraries folder (typically "C:\Program Files (x86)\Arduino\libraries")	
Nore information at : www.objectis-software.com */	
#include "OcfEmbedded.h"	
<pre>int AnalogIn[8]; int DigitalIn[14]; int DigitalOut[14]; int BlinkPeriod[14];</pre>	
<pre>OCFE_BEGIN(ocfEntries) OCFE_VARIABLE_READONLY(AnalogIn[0], OCFE_TYPE_INT) OCFE_VARIABLE_READONLY(AnalogIn[1], OCFE_TYPE_INT) OCFE_VARIABLE_READONLY(AnalogIn[2], OCFE_TYPE_INT) OCFE_VARIABLE_READONLY(AnalogIn[3], OCFE_TYPE_INT) OCFE_VARIABLE_READONLY(AnalogIn[4], OCFE_TYPE_INT) OCFE_VARIABLE_READONLY(AnalogIn[5], OCFE_TYPE_INT) OCFE_VARIABLE_READONLY(AnalogIn[6], OCFE_TYPE_INT)</pre>	۰. ۲
Uploading	
Binary sketch size: 9,108 bytes (of a 258,048 byte maximum)	
1	Arduino Mega 2560 or Mega ADK on COM15



Live Tuning for Arduino with OcfEmbedded

2.5 Test and enjoy with oStudio LiveTuning

- 1. Once the example is uploaded, the Output 13 Led should blink with a 1 second period.
- 2. From the Windows start menu, open "oStudio"
- 3. Select menu File/New/Project



4. In the "Add new project" dialog, select the "Live Tuning" project type, fill in the project name and select the desired folder, then click ok.





Live Tuning for Arduino with OcfEmbedded

5. From the menu "Live Tuning", select "New connection" :

oStu	dic)		_		
File	Live	Tuning Tools Help				
-0	3	New Connection	Section 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1		<u>i</u>	
	0	Add Watch				
	₩	Add Trace				
	103	Add Log	🗵 🗙 😫			
		Add Console	Туре	Path		Value
		Add DropBox				
	检	Add Cs Script				

6. In the "Add New connection" dialog, select "Embedded Com", this is the communication protocol for Ocf Embedded through a serial "Com" port.

oStudio	
File Live Tuning Tools Help	
Add new connection	Value
Sort by categories	
Q Watch	

- 7. In the "Properties" dialog, fill in the appropriate communication parameters :
- Select the "Com" port corresponding to your Arduino.
- Select the baud rate "230400" to match the settings in the example program.
- Click ok.



oStudio		
File Live Tuning Tools Help		
	🍳 🕒 🔍 🔀 🔜 🖿 🛣	
🚓 Home 🗙 🕥 Live Tuning 🗙		
Connections		
	T 🛟 Properties	Value
	Name : OcfeCom	
	Port Name : COM15 - Arduino Mega 2560 (COM15)	
	Baud Rate : 230400 🗸	
	Timeout : 1000 ms	
	O Advanced	
	OK Cancel	

8. Browse the published variables

All the variables published by the example program through the OCFE protocol are exported in the following section of the source code:

S ArduinoOcfe Arduino 1.0.5	
File Edit Sketch Tools Help	
	<mark>.</mark> ₽
ArduinoOcfe	
OCFE_BEGIN(ocfEntries)	A
OCFE_VARIABLE_READONLY(AnalogIn[0], OCFE_TYPE_INT)	
OCFE_VARIABLE_READONLY(AnalogIn[1], OCFE_TYPE_INT)	
OCFE_VARIABLE_READONLY(AnalogIn[2], OCFE_TYPE_INT)	
OCFE_VARIABLE_READONLY(AnalogIn[3], OCFE_TYPE_INT)	
OCFE_VARIABLE_READONLY(AnalogIn[4], OCFE_TYPE_INT)	
OCFE_VARIABLE_READONLY(AnalogIn[5], OCFE_TYPE_INT)	
OCFE_VARIABLE_READONLY(AnalogIn[6], OCFE_TYPE_INT)	_
OCFE_VARIABLE_READONLY(AnalogIn[7], OCFE_TYPE_INT)	=
OCFE_VARIABLE_READONLY(DigitalIn[2], OCFE_TYPE_BOOL)	
OCFE_VARIABLE_READONLY(DigitalIn[3], OCFE_TYPE_BOOL)	
OCFE_VARIABLE_READONLY(DigitalIn[4], OCFE_TYPE_BOOL)	
OCFE_VARIABLE_READONLY(DigitalIn[5], OCFE_TYPE_BOOL)	
OCFE_VARIABLE_READONLY(DigitalIn[6], OCFE_TYPE_BOOL)	
OCFE_VARIABLE_READONLY(DigitalIn[7], OCFE_TYPE_BOOL)	
OCFE_VARIABLE_READONLY(DigitalIn[8], OCFE_TYPE_BOOL)	
OCFE_VARIABLE_READONLY(DigitalIn[9], OCFE_TYPE_BOOL)	
OCFE_VARIABLE(DigitalOut[10], OCFE_TYPE_BOOL)	
OCFE_VARIABLE(DigitalOut[11], OCFE_TYPE_BOOL)	
OCFE_VARIABLE(DigitalOut[12], OCFE_TYPE_BOOL)	
OCFE_VARIABLE(DigitalOut[13], OCFE_TYPE_BOOL)	
OCFE_VARIABLE(BlinkPeriod[10], OCFE_TYPE_INT)	
OCFE_VARIABLE(BlinkPeriod[11], OCFE_TYPE_INT)	
OCFE_VARIABLE(BlinkPeriod[12], OCFE_TYPE_INT)	
OCFE_VARIABLE(BlinkPeriod[13], OCFE_TYPE_INT)	
OCFE_END;	-
4	4

The list of published variables can easily be changed to match your own application. See the Ocfe programmer guide for more information.

All those metadata are read by oStudio when connecting to the Arduino target (or when F5 key is pressed to refresh the tree). So, once connected to the Arduino, oStudio let you immediately browse all those variables with the left tree view.





While the mouse cursor is moving around a variable name, a tooltip appears displaying the current value of the variable, retrieved through OCFE protocol.

9. Monitor the variables

Simply drag drop a variable from the tree on the left to the watch pane on the right. The variable is monitored, and the value is continuously updated.

oStudio			
Eile Live Tuning Iools He Image: State of the state	р <u>©</u> <u>В</u> ×	९ 🔀 🔝 🖬 🖶	
 Gonnections OcfeCom 			
🛹 DigitalOut[10]	Туре	Path	Value
DigitalOut[11] DigitalOut[12]	OcfeCom\Digit	alOut[13]	False
PigitalOut[12]			
A BlinkPeriod[10]			
PlinkPeriod[11]			
BlinkPeriod[12]			
AnalogIn[0]			
AnalogIn[1]			
🚰 AnalogIn[2]			
AnalogIn[3]			
AnalogIn[4]			
AnalogIn[5]			
AnalogIn[7]			

Doing so with the DigitalOutput[13] that is blinking, you will see the toggling of the variable in the watch pane.

10. Remotely change the value of a variable

A double click on any variable will open the "Field invocation" dialog. This dialog will let you type a new value for the variable that will be applied with the "Apply" button.

Live Tuning for Arduino with OcfEmbedded

ostudio			
File Live Tuning Tools Help	0 B Q 🕅 🖻 🖿		
 Connections Connections DigitalOut[10] DigitalOut[11] DigitalOut[12] DigitalOut[13] BinkPeriod[10] BinkPeriod[11] BinkPeriod[12] BinkPeriod[13] AnalogIn[0] AnalogIn[1] AnalogIn[2] AnalogIn[3] AnalogIn[3] AnalogIn[4] AnalogIn[5] AnalogIn[6] AnalogIn[7] DigitalIn[2] DigitalIn[3] DigitalIn[3] DigitalIn[3] DigitalIn[3] DigitalIn[5] DigitalIn[6] DigitalIn[7] DigitalIn[8] DigitalIn[9] 	Field Invocation Type Target : OcteCom\BlinkPeriod[13] Value 500	Image: Second system Evaluate FS Apply F6 Close	Value

To test this functionality, you can change the blink period of the digital output 13 by setting a new value to BlinkPeriod[13] (for example, set the value 500 to set the blink period to 500 ms). In this Arduino ocfe example application, setting the blink period to 0 will stop blinking.

In the same way, you can write the value of a DigitalOutput, by writing the corresponding variable.

11. Tracing a variable

A right click in the tabs will let you create new panes.



Create a new trace pane, and simply drag-drop the variables to trace into this new pane.

Live Tuning for Arduino with OcfEmbedded

oStudio		
<u>F</u> ile Live Tuning <u>T</u> ools <u>H</u> elp		
🔅 Home 🗙 🔘 Live Tuning* 🗙		
Connections M OcfeCom		
DigitalOut[10] DigitalOut[11]	Global	A I
≁ DigitalOut[12] ≁ DigitalOut[13]	Range X:	
BlinkPeriod[10] BlinkPeriod[11]		
BlinkPeriod[12]	Uttset X:	
Analogin[0]		
Analogin[2]	Тгасе	
AnalogIn[4]	ieCom\Digital	Out[13]
AnalogIn[7]	C CreCom\Ana	logIn[0]
TigitalIn[3]	Range Y: 20.0	
♂ DigitalIn[5] ♂ DigitalIn[6]	-5.0 0.0 Offset Y: -3	•
♂ DigitalIn[7] ♂ DigitalIn[8]	Display Configuration	
₿ [™] DigitalIn[9]	🧏 Watch 💆 Trace	

In the picture above, the variables DigitalOut[13] and AnalogIn[0] are plot.

oStudio let you organize your panes in multiple views :

oStudio	
File Live Tuning Tools Help	
Connections Connec	Type Path Value CcfeCom\DigitalOut[13] True Add Image X: Image X: Image X: <t< td=""></t<>

12. Scripting

oStudio offers many additional functionalities, that are extremely useful to tune an application. We will finish by writing a small C# script demonstrating how to write a variable.



Live Tuning for Arduino with OcfEmbedded

Simply create a new pane "CsScript", which let you write some C# code that will be compiled on the fly.

To read or write a variable into the Arduino using Ocfe, simply use the path of the variable and read it or write it as usual with the "=" operator. To make this more easy, you can also simply drag drop a variable from the tree to the script pane.

As an example, we will write a script that slowly changes the blinking period from 0 to 500 ms for output 13.



Click on the run button. The blink period will slowly increase from very fast to normal blinking. The result is cleary visible on the Trace below :





Keep in mind that Ocfe is dynamic typed, so it is sometimes better or even required with C# scripts to use type cast to ensure proper execution. The code written as scripts in oStudio can also be integrated into your own plain .Net applications. To do so, you simply need to download the OcfApi, an extremely easy to use library.

3 More to do with oStudio and LiveTuning

Our experience is that oStudio, the live debugging tool, dramatically improves the development efficiency and the tuning of embedded systems. We wish that you will also get a lot of benefits from oStudio in your own projects.

You can go further with Ocf solutions :

- OcfNative is a communication protocol that let you monitor in the same easy way any of your plain .Net applications. Simply publish your objects with a few lines of code and oStudio let you grab into your application.
- "OcfNative AES encrypted" let you achieve the same results with a high level of security on open networks.
- OcfApi let you very easily integrate Ocf protocols into your own .Net applications in a snap. That way, Ocf is not only a debugging facility, but a first class communication solution.

Ocf is a simple and efficient object oriented communication solution that simplifies to the extreme the interoperability of applications.

OBJECTIS is continuously working on improving oStudio and Ocf to deliver excellent solutions to the community. We will appreciate your feedbacks.

