



<u>US-ARRAY</u>

Programming user guide

(Win dll, matlab, labview)

<u>V1.0</u>





Init_USB

Description :

Initialise the USB3 link

<u>Use</u>:

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Init_USB(Device_Number.l)

Device_Number.l : set the number of usb device (0 for one us-array)

With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

[Retour] = calllib('MyDLL','Init_USB',Device_Number);end





With Labview

🔁 Appeler u	ne fonction d'u	ne DLL			Х
Fonction	Paramètres	Callbacks	Détection d'erreur		
Nom ou C:\US-C D Spéci Nom de Init_US	chemin de la b Compact\US-Co fier le chemin s la fonction B	ibliothèque mpact.dll ur le diagram	me	Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread Convention d'appel stdcall (WINAPI) C	
Prototype d	e la fonction				
uint8_t Init_	USB(uint32_t De	vice_Number);		
				OK Annuler Aide	2





Fonction	Paramètres	Callbacks	Détection d'erreur	
Retour Device	Number	^ ★ ★	Paramètre actuel Nom Retour Type Numérique Constante Type de données Entier 8 bits non signé	
Prototype de uint8_t Init_(e la fonction JSB(uint32_t De	vice_Number); OK Annuler Aid	le

🔛 Appeler une fonctio	on d'une DLL	×
Fonction Paramè	tres Callbacks	Détection d'erreur
Retour Device_Number		Paramètre actuel Nom Device_Number Type Numérique v Constante Type de données Entier 32 bits non signé v Passer Valeur v
Prototype de la foncti uint8_t Init_USB(uint3	ion 32_t Device_Numbe	r); OK Annuler Aide











Open_USB



Open the USB3 link

<u>Use</u>:

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Open_USB(Device_Number.l)

Device_Number.l : set the number of usb device (0 for one us-array)

With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

[Retour] = calllib('MyDLL','Open_USB',Device_Number);end

With Labview



onction	Paramètres	Callbacks	Détection d'erreur		
Nom ou C:\US-C	i chemin de la b Compact\US-Co	ibliothèque ompact.dll			Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread
Nom de	tier le chemin s la fonction JSB	ur le diagram	ime	~	Convention d'appel
					 stdcall (WINAPI) C
totype d	e la fonction	De la Nue			
it8_t Ope	n_USB(uint32_t	Device_Num	ber);		

🔛 Appeler ur	ne fonction d'ur	ne DLL		×
Fonction	Paramètres	Callbacks	Détection d'erreur	
Prototype de	Number		Paramètre actuel Nom Retour Type Numérique Constante Type de données Entier 8 bits non signé	
	1_030(011132_11	Jevice_Ivuini	уст <i>ј</i> ,	
			OK Annuler Aide	





📴 Appeler u	ne fonction d'ur	e DLL			×
Fonction	Paramètres	Callbacks	Détection d'erreur		
Prototype de uint8_t Open	Number e la fonction n_USB(uint32_t [◆ Pevice_Numb	Paramètre actuel No Ty Constan Type de donné Pass er);	m Device_Number De Numérique v te es Entier 32 bits non signé v er Valeur v	
				OK Annuler	Aide
	0	US-Com	npact.dll:lnit_USB	US-Compact.dll:Open_USB	





Close_USB



Close the USB3 link

Use :

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Close_USB()

Device_Number.l : set the number of usb device (0 for one us-array)

With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

[Retour] = calllib('MyDLL','Close_USB');end





Init_Device



Initialise the USB3 link

<u>Use</u>:

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Init_Device(Device_Number.l)

Device_Number.l : set the number of usb device (0 for one us-array)

With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

[Retour] = calllib('MyDLL',Init_Device',Device_Number);end

With Labview





Appeler u	ne fonction d'u	ne DLL		
Fonction	Paramètres	Callbacks	Détection d'erreur	
Nom ou	ı chemin de la b	oibliothèque		
C:\US-0	Compact\US-Co	ompact.dll		
Spéci	ifier le chemin s	ur le diagram	me	
Nom de	e la fonction			
Init_De	vice		~	Convention d'appel
				🔿 stdcall (WINAPI)
				● C
int8_t lnit_	Device(uint32_t	Device_Num	ber);	OK Annuler Aide
int8_t Init_ Appeler u	Device(uint32_t	Device_Num	ber); 	OK Annuler Aide
Appeler u	Device(uint32_t ine fonction d'u Paramètres	Device_Num ine DLL Callbacks	ber); 	OK Annuler Aide
Appeler u Fonction	Device(uint32_t ine fonction d'u Paramètres	Device_Num ine DLL Callbacks	ber); Détection d'erreur Paramètre actuel	OK Annuler Aide
Appeler u Fonction	Device(uint32_t ine fonction d'u Paramètres _Number	Device_Num	ber); Détection d'erreur Paramètre actuel	OK Annuler Aide
Appeler u Fonction Retour Device	Device(uint32_t ine fonction d'u Paramètres _Number	Device_Num	Détection d'erreur Paramètre actuel	OK Annuler Aide
Appeler u Fonction	Device(uint32_t ine fonction d'u Paramètres _Number	Device_Num	ber); Détection d'erreur Paramètre actuel	OK Annuler Aide
Appeler u Fonction	Device(uint32_t ine fonction d'u Paramètres _Number	Device_Num	ber); Détection d'erreur Paramètre actuel Const Type de don	OK Annuler Aide

ОК

Annuler

Aide

uint8_t Init_Device(uint32_t Device_Number);





🔯 Appeler une fonction d'une DLL \times Fonction Paramètres Callbacks Détection d'erreur Paramètre actuel Retour ٨ Device_Number Nom Device_Number + Type Numérique \sim Constante 🗌 Type de données Entier 32 bits non signé Ŷ \sim Passer Valeur \sim Prototype de la fonction uint8_t Init_Device(uint32_t Device_Number); ОК Annuler Aide









GAIN

Description :

Set the amplifiers gains to a selected value. Up to 128 values per channel can be programmed to fit with the 128 sequences.

Features :

Range : 0 to 80 dB Step : 0.1 dB

Use :

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Gain(Device_Number.l,Channel.a,*Tab_In)

Device_Number.l : set the number of usb device (0 for one us-array)

Channel.a : set the channel number (1 to 32)

Tab In : A table filled with a maximum of 128 gain values corresponding to the 128 sequences. Each gain value is a 32 bits Float.

With Matlab





Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

for j=1:64

Tab_Gain(j) = 20; % Gain by sequence

end

for i=1:32

```
Channel = i-1; % Channel selection
```

```
[Retour] = calllib('MyDLL','Gain',Device_Number,Channel,Tab_Gain);
```

end





With Labview

Fonction	Paramètres	Callbacks	Détection d'erreur			
Nom ou C:\US-C D Spécif Nom de Gain	chemin de la b compact\US-Cc fier le chemin s la fonction	ibliothèque mpact.dll ur le diagram	me	-	Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread Convention d'appel stdcall (WINAPI) C	
rototype de iint8_t Gaini	e la fonction (uint32_t Device	e_Number, ui	nt8_t Channel, float *Tab_	_ln);	OK Annuler Ai	de

🔛 Appeler une	fonction d'un	e DLL			×
Fonction	Paramètres	Callbacks	Détection d'erreur		
Retour Device_N Channel Tab_In	umber	^ ∳∳ \$	Paramètre actuel Nom Type Constante Type de données Passer	Channel Numérique	, ii
Prototype de la uint8_t Gain(ui	a fonction int32_t Device_	Number, uir	ıt8_t Channel, float *Tab_ln);	OK Annuler Aid	e



Retour Paramètre actuel Device_Number Nom Tab_In Channel Type Tableau Tab_In Constante Image: Constante Image: Constante
Format du tableau Pointeur sur les données du tableau V







Sampling_Delay

Description:

Set a constant delay on all receivers. The value is apply for all the sequences at the same value.

Features :

Range : 0 to 800 μ S

 $Step: 0.0125 \ \mu S$

<u>Use :</u>

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Sampling_Delay(Device_Number.l,Delay.u)

Device_Number.l : set the number of usb device (0 for one us-array)

Delay.u : value from 0 to 65535, (delay= 0.0125 * Delay.u). Value is unsigned 16





With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

Sampling_Delay = 0;

% Global scanning delay (80 MHz step value)

[Retour] =

calllib('MyDLL','Sampling_Delay',Device_Number,Sampling_Delay);





With Labview

🔁 Appeler un	e fonction d'u	ne DLL				×
Fonction	Paramètres	Callbacks	Détection d'erreur			
Nom ou C:\US-C	chemin de la b ompact\US-Co ïer le chemin s	ibliothèque ompact.dll ur le diagram	me		Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread	
Samplin	ig_Delay			~	Convention d'appel o stdcall (WINAPI) C	
Prototype de uint8_t Samp	la fonction bling_Delay(uin	t32_t Device_	Number, uint16_t Dela	ау);		
					OK Annuler Aid	e



Fonction	Paramètres	Callbacks	Détection d'erreur
Retour		^	Paramètre actuel
Device Delay	Number		Nom Device_Number
			Type Numérique 🗸
			Constante 🗌
			Type de données Entier 32 bits non signé 🗸 🗸
		÷	Passer Valeur 🗸
		*	
ototype d	e la fonction		
nt8_t Sam	pling_Delay(uin	t32_t Device_	Number, uint16_t Delay);

🎦 Appeler u	ne fonction d'ur	ne DLL	×
Fonction	Paramètres	Callbacks	Détection d'erreur
Retour Device Delay	Number	^ ★ ₹	Paramètre actuel Nom Delay Type Numérique Constante Type de données Entier 16 bits non signé Passer Valeur
Prototype d uint8_t Sam	e la fonction pling_Delay(uin	t32_t Device_	Number, uint16_t Delay); OK Annuler Aide











Receiver_Delays

Description :

Set receivers delays for focusing. A table containing up to 128 values is sent on each channel. First value will be used for sequence 1, second value for sequence 2,..... to 128 sequences max.

Features :

Range : 0 to 800 μS Step : 0.0125 μS

<u>Use :</u>

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

*Receiver_Delays(Device_Number.l,Channel.a,*Tab_In)*

Device_Number.l : set the number of usb device (0 for one us-array)

Channel.a : set the channel number (1 to 32)

*Tab_In : Table of delays for all sequences. Each value is unsigned 16.





With Matlab

Declaration :

```
loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-
Compact.h','alias','MyDLL');
```

Calling the function :

for i=1:32

Channel = i-1;

%Channel selection

 $Tab_{In}(1) = 0;$ % Reception delay value for sequence No. 0 (step value = 80 MHz)

[Retour] = calllib('MyDLL','Receiver_Delays',Device_Number,Channel,Tab_In); end





With Labview

🔛 Appeler un	e fonction d'u	ne DLL				\times
Fonction	Paramètres	Callbacks	Détection d'erreur			
Nom ou C:\US-C D Spécif Nom de Receive	chemin de la b ompact\US-Co ïer le chemin s la fonction r_Delays	ibliothèque mpact.dll ur le diagram	me		Thread • Exécuter dans le thread IU • Exécuter dans n'importe quel thread Convention d'appel • stdcall (WINAPI) • C	
Prototype de uint8 t Recei	la fonction ver Delays(uin	t32 t Device	Number, uint8 t Chan	inel, uint16	t*Tab In):	
					OK Annuler Aid	e





Fonction	Paramètres	Callbacks	Détection d'erreur	
Retour Device Channe Tab_In	Number !	^ + × ◆	Paramètre actuel Nom Device_Number Type Numérique Constante Type de données Entier 32 bits non signé Passer Valeur	
rototype de int8_t Rece	e la fonction iver_Delays(uin	t32_t Device_I	Number, uint8_t Channel, uint16_t *Tab_ln); OK Annuler Aic	le

Appeler une fonction d'une DLL	>	<
Fonction Paramètres Callbacks	Détection d'erreur	ł
Retour Device_Number Channel Tab_In X	Paramètre actuel Nom Channel Type Numérique Constante Image: Constante Type de données Entier 8 bits non signé Passer Valeur	-
Prototype de la fonction uint8_t Receiver_Delays(uint32_t Device_	Number, uint8_t Channel, uint16_t *Tab_ln); OK Annuler Aide]



Fonction	Paramètres	Callbacks	Détection d'erreur		
Retour		^	Paramètre actuel		
Device_ Channe	Number	—	Nom	Tab_In	
Tab_In			Туре	Tableau	\sim
			Constante		
		Ŷ	Type de données	Entier 16 bits non signé	\sim
		₽	Dimensions	1	
			Format du tableau	Pointeur sur les données du tableau	\sim
		~	Taille minimale	<none></none>	\sim
ototype d	e la fonction	+22 + Device	Number wint? + Channel wint16 (t *Tab lol:	
no_t kece	iver_Delays(un	toz_t Device_	Number, unito_t channel, unitro_	t Tab_m),	







Transmitter_Delays

Description :

Set transmitters delays for focusing. A table containing up to 128 values is sent on each channel. First value will be used for sequence 1, second value for sequence 2,..... to 128 sequences max.

Features :

Range : 0 to 800 μS Step : 0.0125 μS

<u>Use :</u>

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

*Transmittter_Delays(Device_Number.l,Channel.a,*Tab_In)*

Device_Number.l : set the number of usb device (0 for one us-array)

Channel.a : set the channel number (1 to 32)

*Tab_In : Table of delays for all sequences. Each value is unsigned 16.





With Matlab

Declaration :

```
loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-
Compact.h','alias','MyDLL');
```

Calling the function :

for i=1:32

Channel = i-1;

%Channel selection

 $Tab_{In}(1) = 0;$ % Transmitter delay value for sequence No. 0 (step value = 80 MHz)

[Retour] = calllib('MyDLL','Transmitter_Delays',Device_Number,Channel,Tab_In); end





With Labview

😰 Appeler un	e fonction d'u	ne DLL				Х
Fonction	Paramètres	Callbacks	Détection d'erreur			
Nom ou C:\US-C Spécif Nom de Transm	chemin de la b ompact\US-Cc ïer le chemin s la fonction itter_Delays	ibliothèque mpact.dll ur le diagram	me		Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread Convention d'appel stdcall (WINAPI) C	
Prototype de	la fonction					
uint8_t Trans	smitter_Delays(uint32_t Devi	ce_Number, uint8_t Cl	hannel, uin	t16_t *Tab_ln);	
					OK Annuler Aid	e



Fonction	Parametres	Callbacks	Détection d'erreur		
Retour		^	Paramètre actuel		
Device_ Channe	Number el	F	Nom	Device_Number	
Tab_In			Туре	Numérique	\sim
			Constante		
		Ŷ	Type de données	Entier 32 bits non signé	\sim
		₽	Passer	Valeur	\sim
		~			
ototype d	e la fonction				
int8_t Tran	smitter_Delays(uint32_t Devi	ce_Number, uint8_t Channel, uint	16_t *Tab_ln);	

Retour Device_No Channel Tab_In	umber	Callbacks	Détection d'erreur Paramètre actuel Nom Channel Type Numérique Constante Type de données Entier 8 bits non signé Passer Valeur	
Prototype de la uint8_t Transm	a fonction nitter_Delays(u	iint32_t Devic	ce_Number, uint8_t Channel, uint16_t *Tab_ln); OK Annuler Air	de



Lecoeur Electronique – 300, Chemin des Comtois - 45220 CHUELLES - Te	Tel. : +33 (0)2 38 94 28 30 - Fax : +33 (0)2 38 94 29 67
--	---

onction	Paramètres	Callback	Détection d'erreur	
Petour			Paramètre actuel	
Device_ Channe	Number		Nom Tab_In	
Tab_In		×	Type Tableau	\sim
			Constante 🗌	
		Û	Type de données Entier 16 bits non signé	\sim
		4	Dimensions 1	
			Format du tableau Pointeur sur les données du tableau	\sim
		~	Taille minimale <a>None>	\sim
totype d	e la fonction			
t8_t Tran	smitter_Delays(uint32_t De	ice_Number, uint8_t Channel, uint16_t *Tab_ln);	







Transmitters_Waveforms

Description :

This function allows to program waveforms on each transmitter.

For each channel a table is sent to the hardware. It contains the shape of each waveform for all the sequences. This function must be used with the « size_wave » function wich set the lenght of the wave for the sequences.

Example :

size wave = 12 (number of values sent to define the waveforms)

Waveform for sequence 1

value 1 : always 1

value 2 : positive state for Waveform 1-> 1024+ state width (25 ns step)

value 3 : negative state for Waveform 1 -> 2048 + state width (25 ns step)

value 4 : delay after waveform 1 -> delay (25 ns step)

value 5 : delay before waveform 2 -> delay (25 ns step)
value 6 : positive state for Waveform 2-> 1024+ state width (25 ns step)
value 7 : negative state for Waveform 2 -> 2048 + state width (25 ns step)
value 8 : delay after waveform 2 -> delay (25 ns step)





value 9: delay before waveform 3 -> delay (25 ns step)

value 10 : positive state for Waveform 3-> 1024+ state width (25 ns step)

value 11 : negative state for Waveform 3 -> 2048 + state width (25 ns step)

value 12 : always 1

Waveform for sequence 2

value 13 : always 1

value 14 : positive state for Waveform 1-> 1024+ state width (25 ns step)

value 15 : negative state for Waveform 1 -> 2048 + state width (25 ns step)

value 16 : delay after waveform 1 -> delay (25 ns step)

value 17 : delay before waveform 2 -> delay (25 ns step)

value 18 : positive state for Waveform 2-> 1024+ state width (25 ns step)

value 19 : negative state for Waveform 2 -> 2048 + state width (25 ns step)

value 20 : delay after waveform 2 -> delay (25 ns step)

value 21 : delay before waveform 3 -> delay (25 ns step)

value 22 : positive state for Waveform 3-> 1024+ state width (25 ns step)

value 23 : negative state for Waveform 3 -> 2048 + state width (25 ns step)

value 24 : always 1





EXAMPLE 1

Table Values :

 $\begin{array}{c}
1\\
1024 + 10\\
2048 + 10\\
9\\
\end{array}$ $\begin{array}{c}
1\\
1024 + 10\\
2048 + 5\\
10\\
\end{array}$ $\begin{array}{c}
1\\
1024 + 5\\
2048 + 20\\
1\\
\end{array}$



This table generates the following waveform





EXAMPLE 2

Table Values :

1 1024 + 9 2048 +9 1 1 1024+9 2048+9 1 1024+9 2048+9 1

This table generates the following waveform







Features :

Step size : 25 nS

Memory size : 16384 Values for each channel





<u>Use :</u>

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Transmitter_Wave(Device_Number.l,Channel.a,*Tab_In)

Device_Number.l : set the number of usb device (0 for one us-array)

Channel.a : set the channel number (1 to 32)

*Tab_In : Table of waves values for all sequences. Each value is unsigned 16.

With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

j = 0;

Tab_In = ones(1,135168,'uint16');

for i=1:32

Channel = i-1;

% Channel selection

Tab_In(1) = 1; % First Wave





Tab_In(2) = 20+1024; Tab_In(3) = 20+2048; Tab_In(4) = 1;

Tab_In(5) = 1; % second wave Tab_In(6) = 20+1024; Tab_In(7) = 20+2048; Tab_In(8) = 1;

Tab_In(9) = 1; % Third wave Tab_In(10) = 20+2048; Tab_In(11) = 20+1024; Tab_In(12) = 1;

[Retour] =
calllib('MyDLL','Transmitter_Wave',Device_Number,Channel,Tab_In);
Tab_In = ones(1,135168,'uint16');

end





With Labview





	ne fonction d'u	ne DLL			
Fonction	Paramètres	Callbacks	Détection d'erreur		
Nom ou C:\US-C	ı chemin de la b Compact\US-Co	ibliothèque mpact.dll			Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread
Spéci	ifier le chemin s	ur le diagram	me		· · · · · · · · · · · · · · · · · · ·
Nom de	a la fonction				
Transm	itter_Wave			\sim	Convention d'appel o stdcall (WINAPI) © C
rototype d uint8_t Tran	e la fonction smitter_Wave(u	int32_t Device	e_Number, uint8_t Ch	annel, uint1	6_t *Tab_ln);
					OK Annuler Aid
Appeler u	ne fonction d'un Paramètres	ne DLL Callbacks	Détection d'erreur		
Retour Device Channe	_Number el	^ + ×	– Paramètre actuel –	Nom Type Constante	Device_Number Numérique Image: State of the state o
Tab_In		€ €	51-	Passer	Valeur
Tab_In Prototype d	e la fonction smitter_Wave(u	v int32_t Device	e_Number, uint8_t Ch	Passer	Valeur v 6_t *Tab_ln);

User Guide – US ARRAY Page 40



🔁 Appeler ur	ne fonction d'ur	ne DLL			\times
Fonction	Paramètres	Callbacks	Détection d'erreur		
Retour Device_ Channe Tab_In	Number	^ + × ◆	Paramètre actuel Nom Type Constante Type de données Passer	Channel Numérique Entier 8 bits non signé Valeur	
Prototype de uint8_t Tran	e la fonction smitter_Wave(ui	int32_t Device	e_Number, uint8_t Channel, uint1	6_t *Tab_ln); OK Annuler Aid	e



Appeler u	ine fonction d'ui	ne DLL)
Fonction	Paramètres	Callbacks	Détection d'erreur	
Retour Device Chann Tab_In	_Number el	^ × *	Paramètre actuel Nom Tab_In Type Tableau Constante	
		~	Taille minimale None>	
rototype o	le la fonction nsmitter_Wave(u	int32_t Devi	ce Number, uint8 t Channel, uint16 t *Tab In):	





Size_Wave

Description:

Set the length of the transmitter wave (in words) for each sequence.

<u>Use</u>:

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Size_Wave(Device_Number.l,Size.u))

Device_Number.l : set the number of usb device (0 for one us-array)

Size.u : value from 4 to 16384





With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

Size_Wave = 4; % always a multiple of 4 because a transmitter wave is composed with basic waveforms : delay before state / positive state / negative state / delay after state

[Retour] = calllib('MyDLL','Size_Wave',Device_Number,Size_Wave);





With Labview



Appeler ur	ne fonction d'ui	ne DLL		
Fonction	Paramètres	Callbacks	Détection d'erreur	
Nom ou C:\US-C Spéci Nom de Size_Wa	chemin de la b Compact\US-Co fier le chemin si la fonction ave	ibliothèque mpact.dll ur le diagram	me	Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread Convention d'appel stdcall (WINAPI) C
Prototype de	e la fonction			
nt8_t Size_V	Vave(uint32_t D	evice_Numbe	r, uint16_t size wave);	
				OK Annuler Aid





🔁 Appeler u	ne fonction d'ur	ne DLL	>	<
Fonction	Paramètres	Callbacks	Détection d'erreur	
Retour Device size way	Number ve	▲	Paramètre actuel Nom Device_Number Type Numérique Constante Image: Constante Type de données Entier 32 bits non signé Passer Valeur	
Prototype de int8_t Size_V	e la fonction Vave(uint32_t De	evice_Numbe	r, uint16_t size wave); OK Annuler Aide	



🛂 Appeler u	ne fonction d'ui	ne DLL		×
Fonction	Paramètres	Callbacks	Détection d'erreur	
Retour Device <u></u> size wa	_Number ve	^ ★ ↓	Paramètre actuel Nom size wave Type Numérique Constante	
Prototype d int8_t Size_V	e la fonction Wave(uint32_t D	evice_Numbe	er, uint16_t size wave); OK Annuler	Aide





Sampling_Freq

Description :

Set the sampling frequency for the ADC.

1-> 80 MHz

2-> 40 MHz

4-> 20 MHz

8-> 10 MHz

<u>Use</u>:

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Sampling_Freq(Device_Number.l,Freq.u)

Device_Number.l : set the number of usb device (0 for one us-array)

Freq.u : value 1 or 2 or 4 or 8





With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Calling the function :

sampling_Freq = 1; % (1 -> 80 MHz, 2 -> 40 MHz, 4 -> 20 MHz, 8 -> 10 MHz)

[Retour] =
calllib('MyDLL','Sampling_Freq',Device_Number,Sampling_Freq);









netion	Paramètres	Callbacks	Détection d'erreur		
Nom ou C:\US-C	chemin de la b Compact\US-Co fier le chemin s	ibliothèque ompact.dll ur le diagram	me		Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread
Nom de Samplin	la fonction ng_Freq		~	ł	Convention d'appel Stdcall (WINAPI) C
ototype de	e la fonction	22 + Device N	lumber uint16 t sampling	(freg):	



	ine romenom a a	ne DLL		
Fonction	Paramètres	Callbacks	Détection d'erreur	
Retour		•	Paramètre actuel	
Device_ samplir	<u>Number</u> na frea		Nom	Device_Number
			Туре	Numérique 🗸
			Constante	
			Type de données	Entier 32 bits non signé 🗸 🗸
		v	Passer	Valeur
ototype d nt8_t Sam	e la fonction pling_Freq(uint	32_t Device_N	lumber, uint16_t sampling freq);	
				OK Annuler Aide
Appeler u	ne fonction d'u	ine DLL		
Appeler u Fonction	ne fonction d'u Paramètres	ne DLL Callbacks	Détection d'erreur	
Appeler u Fonction	ne fonction d'u Paramètres	Callbacks	Détection d'erreur Paramètre actuel	
Appeler u Fonction Retour Device	ne fonction d'u Paramètres Number	Callbacks	Détection d'erreur Paramètre actuel Nom	sampling freq
Appeler u Fonction Retour Device samplin	ne fonction d'u Paramètres Number ng freq	Callbacks	Détection d'erreur Paramètre actuel Nom Type	sampling freq
Appeler u Fonction Retour Device samplin	ne fonction d'u Paramètres <u>Number</u> ng freq	Callbacks	Détection d'erreur Paramètre actuel Nom Type Constante	sampling freq
Appeler u Fonction Retour Device <u></u> samplin	ne fonction d'u Paramètres Number ng freq	Callbacks	Détection d'erreur Paramètre actuel Nom Type Constante Type de données	sampling freq
Appeler u Fonction Retour Device samplin	ne fonction d'u Paramètres Number ng freq	ne DLL Callbacks	Détection d'erreur Paramètre actuel Nom Type Constante Type de données Passer	sampling freq Numérique Entier 16 bits non signé Valeur
Appeler u Fonction Retour Device samplin	ne fonction d'u Paramètres Number ng freq	ne DLL Callbacks	Détection d'erreur Paramètre actuel Nom Type Constante Type de données Passer	sampling freq Numérique Entier 16 bits non signé Valeur
Appeler u Fonction Retour Device samplin	ne fonction d'u Paramètres Number ng freq e la fonction	ne DLL Callbacks	Détection d'erreur Paramètre actuel Nom Type Constante Type de données Passer	sampling freq Numérique Entier 16 bits non signé Valeur
Appeler u Fonction Retour Device samplin samplin	ne fonction d'u Paramètres Number ng freq e la fonction pling_Freq(uint	Ine DLL Callbacks	Détection d'erreur Paramètre actuel Nom Type Constante Type de données Passer	sampling freq Numérique Entier 16 bits non signé Valeur





Frame_Mux

Description :

Programs the multiplexer 32 -> 64 channels. For Each sequence a 64 bits word is applied at the input of the multiplexer. Each bit drives a switch of the multiplexer. According to the following drawing



A table composed with all the 64 bits words is sent by calling the DLL. A 64 bits word is composed with four 16 bits words (LSB,LSB-1,MSB-1,MSB). To program all sequences (64) the table size is 256 values.





Use :

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Frame_Mux (Device_Number.1,*Tab_In)

Device_Number.1 : set the number of usb device (0 for one us-array

*Tab_In : Table of multiplexer control words (see before for description)

With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

i = 0;

for *i*=1:128

 $Tab_In(i+j) = i;$ % $Tab_{In(i+1+j)} = 0;$ % $Tab_{In(i+2+j)} = 0;$ % *Tab* In(i+3+i) = 0;% i = i + 3;

$$j = j + 3$$

end

[*Retour*] = *calllib*('*MyDLL*','*Frame_Mux*',*Device_Number*,*Tab_In*);





With Labview

Appeler une fonction d'une D	DLL		×
Fonction Paramètres Ca	allbacks Détection d'e	rreur	
Retour Device_Number frame tab	Paramètre a	ctuel Nom Retour Type Numérique Constante Image: Constante Type de données Entier 8 bits non signé	
Prototype de la fonction uint8_t Frame_Mux(uint32_t Dev	vice_Number, uint16_t *f	irame tab); OK Annuler Aide	e



Lecoeur Electronique - 300, Chemin des Comtois - 45220 CHUELLES - Tel. : +33 (0)2 38 94 28 30 - Fax : +33 (0)2 38 94 29 67

	Dararahtura	Callbacks	Détection d'annous			
onction	Parametres	Calibacks	Detection d erreur			
Nom ou	chemin de la l	bibliothèque	C D D		Thread	
Compa	ct.dll	(US_Array\VS	Sources Purebasic U	S- 2	Exécuter dans le thread IU	
					O Exécuter dans n'importe quel thread	
🗌 Spéci	fier le chemin s	sur le diagram	ime			
Nom de	la fonction					
Frame_	Mux			\sim	Convention d'appel	
					🔿 stdcall (WINAPI)	
					● C	
ototype de	e la fonction					
- 10 I E	M. 4. 1. 100	A Desider March	- 1	L-L-1.		
nt8_t Fram	ne_Mux(uint32_	_t Device_Nun	nber, uint16_t *frame	tab);		
nt8_t Fram	ne_Mux(uint32_	t Device_Nun	nber, uint16_t *frame	tab);		
nt8_t Fram	ne_Mux(uint32_	_t Device_Nun	nber, uint16_t *frame	tab);	OK Annuler	Aid
nt8_t Fram	ne_Mux(uint32_	_t Device_Nun	nber, uint16_t *frame	tab);	OK Annuler	Aid
nt8_t Fram	e_Mux(uint32_	_t Device_Nun	nber, uint16_t *frame	tab);	OK Annuler	Aid
Appeler un	ne_Mux(uint32_ e fonction d'u Paramètres	t Device_Nun	nber, uint16_t *frame Détection d'erreur	tab);	OK Annuler	Aid
Appeler un	ne_Mux(uint32_ e fonction d'u Paramètres	_t Device_Nun ne DLL Callbacks	nber, uint16_t *frame Détection d'erreur	tab);	OK Annuler	Aid
Appeler un onction	e fonction d'u	t Device_Nun	nber, uint16_t *frame Détection d'erreur – Paramètre actuel –	tab);	OK Annuler	Aid
Appeler un onction Retour Device_1 frame ta	ne_Mux(uint32_ ne fonction d'u Paramètres Number b	t Device_Nun	nber, uint16_t *frame Détection d'erreur – Paramètre actuel –	tab); Nom	OK Annuler Device_Number	Aid
Appeler un onction Retour Device_I frame ta	e fonction d'u Paramètres Number b	t Device_Nun	nber, uint16_t *frame Détection d'erreur Paramètre actuel –	tab); Nom Type	OK Annuler Device_Number Numérique	Aid
Appeler un onction Retour Device I frame ta	ne_Mux(uint32_ ne fonction d'u Paramètres Number b	t Device_Nun	nber, uint16_t *frame Détection d'erreur Paramètre actuel –	tab); Nom Type Constante	OK Annuler Device_Number Numérique	Aid
Appeler un onction Retour Device_1 frame ta	e_Mux(uint32_ e fonction d'u Paramètres Number b	t Device_Nun	Détection d'erreur Paramètre actuel – Type d	Nom Type Constante de données	OK Annuler Device_Number Numérique ✓ Entier 32 bits non signé ✓	Aid
Appeler un onction Retour Device_f frame ta	ne_Mux(uint32_ ne fonction d'u Paramètres Number Ib	t Device_Nun	Détection d'erreur Paramètre actuel – Type o	Nom Nom Type Constante de données Passer	OK Annuler	Aid
Appeler un onction Retour Device_1 frame ta	ne_Mux(uint32_ ne fonction d'u Paramètres Number b	t Device_Nun	Détection d'erreur Paramètre actuel - Type o	Nom Nom Type Constante de données Passer	OK Annuler OK Annuler Device_Number	Aid

ОК

22 1

Annuler

Aide

uint8_t Frame_Mux(uint32_t Device_Number, uint16_t *frame tab);





onction Paramètres	Callbacks	Détection d'erreur		
Pataur		Paramètre actuel		
Device_Number		Nom	frame tab	٦
frame tab	+	-		
	$\mathbf{\times}$	lype	Tableau	1
		Constante		
	Ŷ	Type de données	Entier 16 bits non signé	/
	₽	Dimensions	1	
		Format du tableau	Pointeur sur les données du tableau	~
	× .	Taille minimale		
ototype de la fonction				
nt8_t Frame_Mux(uint32_	t Device_Num	nber, uint16_t *frame tab);		
			OK Annuler	Aide
		US-Compact.dll:Fr	ame_Mux	

tab in





Data_RF

Description :

This function must be used to acquire RF raws sampled by US-ARRAY CANs. Calling this function returns the RF rows of the 32 Channels. Each RF signal is 4096 samples long. Each sample is a 16 bits (U16) word but the sample is on 10 bits (Six last bits are set to 0) coded from 0 to 1023. The size of the returned table is 4096 * 32 = 131 072 words:

0 - RF channel 1

4096 - RF channel 2

8192 - RF channel 3

•••

•••

126 976 – RF channel 32





<u>Use :</u>

With : DLL US-Compaq (C:\US-Compact\US-Compact.dll)

Data_RF (Device_Number.l,*Tab_out)

Device_Number.l : set the number of usb device (0 for one us-array)

*Tab_out: Table of samples as descripted before (size: 131072).

With Matlab

Declaration :

loadlibrary('C:\US-Compaq\US-Compact.dll','C:\US-Compaq\US-Compact.h','alias','MyDLL');

Offset = ones(1,4096*33,'double')*512; % Tableau utilisé pour centrer le signal sur "0" *tStart* = *tic*; [*Retour Data*] = *calllib*('*MyDLL*','*Data_RF*',*Device_Number*,*Tab_In*); % 32 Channels acquisition % Channel N°2 -> [4097 8192], ...) *tElapsed* = *toc(tStart);* Data = cast(Data,'double'); % uint16 in double Data = Data - Offset; % signal normalisation plot(Data*double(0.002)); % volts conversion xlabel('Samples','FontSize',14); ylabel('Volts','FontSize',14); axis([1 4096 -1 1]); drawnow;





With Labview

🔛 Appeler ur	ne fonction d'u	ne DLL			Х
Fonction	Paramètres	Callbacks	Détection d'erreur		
Nom ou US-Com Spécit Nom de Data_RF	chemin de la b npact.dll fier le chemin s la fonction	ibliothèque ur le diagram	me	Thread Exécuter dans le thread IU Exécuter dans n'importe quel thread Convention d'appel stdcall (WINAPI) C	
Prototype de	e la fonction	vice Number			
uinto_t Data	_rr=(uints2_t De	vice_ivumber	, uint lo_t "Tab_Out);		
				OK Annuler Aid	e



nction	Paramètres	Callbacks	Détection d'erreur			
			– Paramètre actuel –			
Retour	Number	^		г		
Tab_Ou	it	+		Nom	Retour	
				Туре	Numérique	\sim
				Constante [
			Type d	le données	Entier 8 bits non signé	\sim
		Ŧ		l		
		-				
		~				
totype d	e la fonction					
t8_t Data	_RF(uint32_t De	evice_Number	; uint16_t *Tab_Out);			
					OK	Appuler Ai
					ОК	Annuler Ai
opeler un	ne fonction d'ui	ne DLL			OK .	Annuler Ai
opeler un	ne fonction d'u	ne DLL			ОК 4	Annuler Ai
opeler un nction	ne fonction d'ui Paramètres	ne DLL Callbacks	Détection d'erreur		OK	Annuler Ai
opeler un nction	ne fonction d'ur Paramètres	ne DLL Callbacks	Détection d'erreur	- I	OK .	Annuler Ai
opeler un nction Retour	ne fonction d'ur Paramètres	ne DLL Callbacks	Détection d'erreur Paramètre actuel		OK /	Annuler Ai
nction Retour Device_	ne fonction d'un Paramètres Number	ne DLL Callbacks	Détection d'erreur Paramètre actuel	Nom	OK	Annuler Ai
opeler un nction Retour Device_ Tab_Out	ne fonction d'un Paramètres Number t	ne DLL Callbacks	Détection d'erreur Paramètre actuel	Nom	OK /	Annuler Ai
opeler un nction Retour Device_ Tab_Our	ne fonction d'un Paramètres <mark>Number</mark> t	ne DLL Callbacks	Détection d'erreur Paramètre actuel	Nom Type	OK /	Annuler Ai
nction Retour Device_ Tab_Our	ne fonction d'un Paramètres Number t	ne DLL Callbacks	Détection d'erreur Paramètre actuel	Nom Type Constante	OK /	Annuler Ai
ppeler un nction Retour Device_ Tab_Our	ne fonction d'un Paramètres <mark>Number</mark> t	ne DLL Callbacks	Détection d'erreur Paramètre actuel Type d	Nom Type Constante e données	OK C Device_Number Numérique	Annuler Ai
ppeler un nction Retour Device_ Tab_Our	ne fonction d'un Paramètres Number t	ne DLL Callbacks	Détection d'erreur Paramètre actuel Type d	Nom Type Constante e données Passer	OK A	Annuler Ai
ppeler un nction Retour Device_ Tab_Our	ne fonction d'un Paramètres <mark>Number</mark> t	ne DLL Callbacks	Détection d'erreur Paramètre actuel Type d	Nom Type Constante données Passer	OK /	Annuler Ai
ppeler un nction Retour Device_ Tab_Our	ne fonction d'un Paramètres Number t	ne DLL Callbacks	Détection d'erreur Paramètre actuel Type d	Nom Type Constante e données Passer	OK A	Annuler Ai
ppeler un nction Retour Device_ Tab_Out	e fonction d'un Paramètres <mark>Number</mark> t	ne DLL Callbacks	Détection d'erreur Paramètre actuel Type d	Nom [Type [Constante] e données [Passer [OK /	Annuler Ai
ppeler un nction Retour Device_ Tab_Out	e fonction d'un Paramètres Number t	ne DLL Callbacks	Détection d'erreur Paramètre actuel Type d	Nom Type Constante e données Passer	OK A	Annuler Ai
opeler un nction Retour Device_ Tab_Our	e fonction d'un Paramètres Number t	ne DLL Callbacks	Détection d'erreur Paramètre actuel Type d	Nom [Type [Constante] e données [Passer [OK /	Annuler Ai



onction	Paramètres	Callbacks	Détection d'erreur				
onceion		cambacità	beteenion a circar				
Retour		^	Paramètre actuel				
Device_Number		-	Nom Tab_Out				
_	-	×	Туре	Tableau	~		
			Constante 🗌				
		Û	Type de données	Entier 16 bits non signé	~		
		₽	Dimensions	1			
			Format du tableau	Pointeur sur les données du tableau	~		
		~	Taille minimale	<none></none>	~		
ototype d	e la fonction						
nt8_t Data	_RF(uint32_t De	vice_Numbe	; uint16_t *Tab_Out);				

Lecoeur Electronique - 300, Chemin des Comtois - 45220 CHUELLES - Tel. : +33 (0)2 38 94 28 30 - Fax : +33 (0)2 38 94 29 67

acquisition of 32 channels



