# 1 TABLE DES MATIERES

2	Exp	Expérimentation avec le DAC N°1 d'une carte Nucléo-H755ZI						
3	Ini	Initialisation de l'environnement						
4	Со	Conception et codage						
	4.1	Conception	14					
	4.2	Programmation	15					
	4.3	Petite évolution	22					

# 2 EXPERIMENTATION AVEC LE DAC N°1 D'UNE CARTE NUCLEO-H755ZI

Ce tutoriel est une déclinaison du tutoriel « DAC-DMA-TIMER Nucléo-L476RG », mais il est plus direct, dans le sens où nous irons directement au but correspondant à l'étape 3.

La mise en œuvre du microcontrôleur STM32H755ZI est un véritable enfer ! Le résultat ici présenté est le fruit de nombreuses recherches et investigations que je passerai sous silence.

Défi : vous recherchez un bon passe-temps ? Si oui, ne lisez pas ce tutoriel et essayez de générer par vous-même un squelette de programme via STM32CubeMx dans le but de mettre en œuvre {DAC+TIMER2+DMA} et de créer une sinusoïde de fréquence variable.

La suite de ce tuto représente la solution à ce défi dans l'environnement MDK-ARM de  $\mu Vision.$ 

## 3 INITIALISATION DE L'ENVIRONNEMENT

STM32CubeMX peut grandement soulager l'utilisateur en générant le maximum de code à sa place, il suffit pour cela de lui indiquer nos choix en matière de :

- carte Nucléo, ici ce sera le modèle Nucléo-H755ZI ;
- préférences pour la génération automatique de code ;
- périphérique à mettre en œuvre, ici ce sera le DAC1 ;
- fréquence d'horloge, on prendra la fréquence par défaut proposée par STM32CubeMX : 64 MHz ;
- nom et endroit de stockage du projet.
- Choix de l'IDE, ici ce sera MDK-ARM.

Voici comment lui indiquer ce que nous souhaitons :

- 1) Lancer STM32CubeIDE.
- 2) FILE/NewProject.
- 3) Choisir l'onglet « Board Selector »
- 4) Dans « Part Number Search », choisir NUCLEO-H755ZI-Q.
- 5) Cliquer dans la fenêtre de droite sur la carte Nucléo correspondante.
- 6) Cliquer sur le bouton « Start Project ».
- 7) « Initialize all peripherals with their default Mode » → répondre Yes
- 8) Dans l'onglet « Pinout & Configuration »
  - a. Développer Analog et cliquer sur DAC1
    - i. Dans la partie Mode
      - 1. Dans Runtime contexts : cocher uniquement « Cortex-M7 »
      - 2. Out1 mode choisir Connected to external pin only.
    - ii. Dans la partie Configuration
      - 1. Cliquer sur l'onglet Parameter Settings → dans la partie pinout view vérifier que DAC1\_OUT1 est assigné à PA4.
      - 2. Trigger choisir Timer 2 Trigger Out event.

STM32CubeM	IX Untitled*: STM32H755ZITx	NUCLEO-H755ZI-Q					- a ×
STM32 CubeMX		File	Window	Help			🧐 🖪 🖻 🎽 🔆 🏹
Home	STM32H755ZITx - NU	JCLEO-H755ZI-Q	> Untitled - Pinc	ut & Configuration >			GENERATE CODE
	Pinout & Configu	ıration	С	lock Configuration		Project Manager	Tools
				Additional Software	✓ Pinout		
Q	✓ () A₂>7			DAC1 Mode and Configuration			Pinout view III System view
System Co	ore >	Runtime contex Corte	ts: 8 x-M7	) a.i.1. Cortex-M4	PowerDomain		
Analog	~	OUT1 mode Conr	ected to external pin	only 8) a i 2	D2	×	
ADC1	÷	OUT2 mode Disa	ble	C/ uniz:		✓ I	
ADC2		🗆 External Trigg	er			¢.	
	( 8) a					_	STM32H755ZI
COMP	2		_	Configuration		2	• • • • • • • • • • • • • • • • • • • •
V DAC1	04	Reset Configurati	8) a.ii.1				
OPAM	P2	Parameter Settin	ngs 😔 User Const	ants 🛛 📀 NVIC Settings 👘	⊙ DMA Settings │ ⊙ GPIO S	iettings	LQFP144
VREFE	BUF	Configure the below p	arameters :				
		Q Search (CrtI+F)					
Timers	>	Output Bu	ffer <b>8) a.</b> l	i.2	Enable		
Connectivit	ty >	Trigger Wave gene	eration mode		Timer 2 Trigger Out event Disabled		
Multime dia		User Trimr	ning		Factory trimming		
Multimedia	· · · · ·	Sample Ar	nd Hold		Sampleandhold Disable		
Security	>					H_MBR	d_SAT A_RXC Hen Leo
Computing	>					ਸ਼ੂੰ <sup>⊑</sup> 8) a.ii.1 ੁੱ	티프 이 Gee 티프
	. 、						3

- 9) Partie « Configuration » du DAC1 », dans l'onglet « DMA settings » :
  - a. Cliquer sur le bouton « Add » ;
  - b. Dans la boîte « select » qui s'ouvre, choisir « DAC1\_CH1 »
  - c. Dans la partie « DMA Request Settings »
    - i. choisir le Mode « Circular »
    - ii. Data Width : « Word » (du côté Peripheral)
    - iii. Data Width : « Word » (du côté Memory).

STM32 CubeMX	F	ile	Window	ŀ	Help		
Home 🔰 STM32H758	5ZITx - NUCL	.EO-H755ZI-Q	Untitled - Pir	nout & Conf	iguration >		
Pinout &	Configura	tion		Clock Cor	nfiguration		Proje
				Additional	Software	$\sim$	Pinout
Q ~	٢			DAC1 Mode	and Configuration		
Categories A->Z				١	Node		
System Core	>	Runtime context	S: • M7	Cor	tov MA	Pow	orDomain.
Angles	~		K-1V1/	00		FOW	D2
Analog		OUT1 mode Conn	ected to external pi	in only			~
€ ADC1		OUT2 mode Disab	le				~
ADC2		External Trigge	er				
ADC3	- 11			Conf	iguration		
COMP1		Reset Configuratio	n			9)	
V DAC1	_	Daramatar Satting	an 🖉 Hear Can	etente 🔼 🗛	N/C Sattings	DMA Sottings	C CPIO Settingo
OPAMP1			gs   🔮 Oser Con	stants 🕑 i	avic Settings	UMA Settings	GPIO Settings
OPAMP2	- H	DMA Reques	t S	Stream	Dire	tion	Priority
VREFBUF		DAC1_CH1	✓ DMA1 Stre	am 0	Memory To Pe	eripheral Low	1
		Add Del	ete 37 D.				
Timers	>	DMA Request Setti	ings				
Connectivity	>	9) c.i	i.			Peripheral	Memory
connocinity		Mode Circular	~		Incromont Addr		
Multimedia	<u> </u>	Mode Circular	· ·				
Security	>	Use Fifo 🔲 Th	reshold	$\sim$	Data Width	Word	∼ Word ~
Computing	>				9 Burst Size	) c.ii	─ 9) c.iii _
Middleware	>	DMA Request Synd	chronization Settin	gs			
		Enable synchroniz	ation				

10) Onglet Clock Configuration

a. Vérifier les fréquences à 64 MHz comme dans l'arbre d'horloge ci-dessous.



11) Il reste à configurer le timer2 que nous avons choisi :

- a. Partie « Pinout & Configuration » → choisir Tim2.
- b. Partie « Tim2 Mode & Configuration / Mode », faire les réglages suivants :
  - i. Dans Runtime contexts : cocher uniquement « Cortex-M7 »
  - ii. « Clock Source » → « Internal Clock » ;
- c. Partie « Tim2 Mode & Configuration /Configuration/Parameter Settings », faire les réglages suivants :
  - i. Prescaler (PSC 16 bit value) → 64-1 Ceci a pour effet de diviser la fréquence d'horloge par 64, ce qui fait qu'en sortie du prescaler nous aurons 1 MHz en guise de fréquence d'horloge pour le timer2.
  - ii. → « Counter Period (AutoReload Register -32 bit value) » → 100-1. Ceci a pour effet de générer une interruption tous les 100 coups d'horloge et donc d'obtenir une fréquence de 10 kHz pour le DMA.
  - iii.  $\rightarrow$  « Trigger Event Selection »  $\rightarrow$  « update event ».



STM32 CubeMX		File		Window	Н	lelp			
Home >	STM32H755ZIT	Tx - NUCLEO	-H755ZI-Q	➢ Untitled - P	inout & Confi	guration >			
	Pinout & C	onfiguratior	1		Clock Con	figuration			Proj
					Additional	Software	~	Pinout	
Q	$\sim$	٢			TIM2 Mode a	nd Configuration	11) b.		
Categories	A->Z				N	lode			
Analog		~	Runtime conte	11) b.i.					
Timers		~	Cort	ex-M7	Cort	ex-M4	Pow	erDomain	
		— II L		×				D2	
HDTIN	•	S	Blave Mode Disa	ble					~
LPTIM	1	1	rigger Source D	isable					$\sim$
LPTIM		(	lock Source	Internal Clock	11) b.ii				$\sim$
LPTIM	3				Confi	guration			
LPTIM	4		Denot Configurat	ion					
LPTIM			keset Conligurat	11) c.				_	
RTC	11) a.		Parameter Setti	ngs 🕑 User Co	onstants 🛛 📀 N	VIC Settings 🛛 😔	DMA Setting	5	
11101		Conf	gure the below	parameters :					
		<b>Q</b> S	earch (Crtl+F)	0					0
TIM4		~ (	Count <u>er Settings</u>						
TIM5			Prescaler	(PSC - 16 bits valu	ue)	64-1 11) c.i			
TIM6			Counter N	Aode	Decision 22	Up			
TIM7			Internal C	eriod (AutoReload) lock Division (CKD)	Register - 32	100-1 11) C.I			
TIM8			auto-reloa	id preload	)	Disable			
TIM12		✓ T	rigger Output (T	RGO) Parameters					
TIM13			Master/S	ave Mode (MSM bi	it)	Disable (Trigger in	put effect not	lelaved)	0
TIM14			Trigger Ev	ent Selection TRG	0	Update Event	11) c.iii		о́ 

d. → Dans l'onglet « NVIC Settings » cocher « Enabled » en face de « TIM2 blobal interrupt ».

Pinout	& Configura	tion	Clock C	onfiguration		F
			Addition	al Software	🗸 Pi	nout
~	0		TIM2 Mod	e and Configuration		
tegories A->Z				Mode		
Analog	>	Runtime contexts:				
imore	~	Cortex-M7	C	ortex-M4	PowerD	omain
lillers		✓			D2	
\$		Slave Mode Disable				~
HRIIM		Trigger Source Disable				~
LPTIM1		Clock Source Internal Cloc	ck			~
LPTIM3			C	onfiguration		
LPTIM4				Inguration		
LPTIM5		Reset Configuration				
RTC		🥺 Parameter Settings 🛛 📀 l	Jser Constants 🛛 😔	) NVIC Settings 🛛 🧕	DMA Settings	
TIM1		NVIC1 Interrupt Ta	able	Enabled Pree	mption Priority	Sub Priority
/ TIM2		TIM2 global interrupt		<b>0</b>		0
TIM3						
TIM4			11) d.	ů		

- 12) Onglet Project Manager/Project
  - a. Nommer le projet
  - b. Donner l'endroit où stocker le projet
  - c. choisir l'IDE Toolchain/IDE: MDK-ARM pour ce qui me concerne.

STM32	File	Window	Help
Home >	STM32H755ZITx - NUCLEO-H755ZI-Q 🔰 (	Untitled - Project Man	ager 🔪
	Pinout & Configuration	Clock C	onfiguration
<b>12)</b> Proje	Project Settings Project Name Exp-DAC-DMA-TIMER Project Location D:\Documents\Informatique\Programmes' Dual Core Boot Mode	\STM32\NUCLEO-H755ZI-(	12) a. 21 12) b. Browse
Code Ge	enerator Both CPUs booting at once Application Structure Basic Toolchain Folder Location matique\Programmes\STM32\NUCLEO-H Toolchain (JDE Min Vo)	Do not generation-1	ate the main() DAC-DMA-TIMER-Nucléo-H755ZI\
Advanced	Linker Settings Minimum Heap Size 0x200 Minimum Stack Size 0x400	12) c.	Generate Under Root
	Mcu and Firmware Package Mcu Reference STM32H755ZITx Firmware Package Name and Version		

STM32CubeMX Untitled\*: STM32H755ZITx NUCLEO-H755ZI-Q

- 13) Onglet Project Manager/Code Generator
  - a. Onglet horizontal « Code Generator » → cocher « Generate peripheral initialization as a pair of '.c/h' files per peripheral »

STM32CubeMX Untitled\*: STM32H755ZITx NUCLEO-H755ZI-Q

STM32 CubeMX	File	Window	Help			
Home > STM32H	755ZITx - NUCLEO-H755ZI-Q	Vntitled - Project	ct Manager >			
Pinout	& Configuration	Clo	ock Configuration			
Project	STM32Cube MCU packages and en Copy all used libraries into the p Copy only the necessary library Add necessary library files as re	nbedded software packs oroject folder files iference in the toolchain	project configuration file			
13) Code Generator	Generated files       Image: Constant of the ser peripheral initialization as a pair of the ser peripheral initialization as					
Advanced Settings	HAL Settings Set all free pins as analog (to op Enable Full Assert Template Settings Select a template to generate custor	ntimize the power consu	mption) Settings			

#### 14) Fenêtre principale de STM32CubeMX

- a. Faire « FILE/SAVE »
- b. Appuyer sur le bouton Generate Code.



c. Dans le popup qui s'affiche appuyer sur le bouton « Open Project »...

- 15) L'IDE Keil µVision s'ouvre, il faut changer les paramètres :
  - a. Choisir d'abord la cible CM4

D:\Documents\Informatique\Programmes\STM32\NUCLEO-H755ZI-Q\Exp-DAC-DMA-TIMER\MDK-ARM\Exp-DAC



b. Ouvrez les options du projet avec ALT+F7 et vérifiez scrupuleusement les écrans suivants :

Options for Target 'Exp-DAC-DMA-TIMER_CM4'     ×							
Device Target Output Listing User C/C++ Asm Linker Debug Utilities							
STMicroelectronics STM32H755ZITx:CM4	Code Generation						
<u>⊠</u> tal (MHz): 64.0	Anni Complier. Use derault complier vers						
Operating system: None	Use Cross-Module Optimization						
System Viewer File:	Use MicroLIB Eig Endiar						
Use Custom File	Stoating Foint Haleware. Single Precision						
Read/Only Memory Areas	Read/Write Memory Areas						
default off-chip Start Size Startup	default off-chip Start Size	Nolnit					
□ ROM1: □ C	RAM1:						
□ ROM2: C	RAM2:						
C ROM3.	RAM3:						
IROM1: 0x8100000 0x100000	IRAM1: 0x1000000 0x48000						
IROM2:	IRAM2:						
ОК Са	ncel Defaults	Help					
Options for larget 'Exp-DAC-DMA-TIMER_CM4'	X						
Device   Target Output   Listing   User   C/C++   Asm   Linker	Debug   Utilities						
Select Folder for Objects Name of Executable: Exp-DAC-DMA-TIMER_CM4							
	Exp-DAC-DMA-TIMER_CM4						
Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA	-TIMER_CM4						
Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA ✓ Debug Information	-TIMER_CM4						
Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA     Debug Information     Create HEX File     Browse Information	•: Exp-DAC-DMA-TIMER_CM4						
	ER_CM4.						
Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA     Debug Information     Create HEX File     Browse Information     Create Library: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA-TIMER_CM4-Exp-DAC-DMA	ER_CM4.lib						
<ul> <li>Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA</li> <li>✓ Debug Information</li> <li>✓ Create HEX File</li> <li>✓ Browse Information</li> <li>C create Library: Exp-DAC-DMA-TIMER_CM4\Ex</li></ul>	ER_CM4.lib						
<ul> <li>Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA</li> <li>☑ Debug Information</li> <li>☑ Create HEX File</li> <li>☑ Browse Information</li> <li>☑ Create Library: Exp-DAC-DMA-TIMER_CM4\Ex</li></ul>	ER_CM4.lib						
<ul> <li>Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA</li> <li>☑ Debug Information</li> <li>☑ Create HEX File</li> <li>☑ Browse Information</li> <li>☑ Create Library: Exp-DAC-DMA-TIMER_CM4\Ex</li></ul>	ER_CM4.lib						
<ul> <li>Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA</li> <li>☑ Debug Information</li> <li>☑ Create HEX File</li> <li>☑ Browse Information</li> <li>☑ Create Library: Exp-DAC-DMA-TIMER_CM4\Ex</li></ul>	ER_CM4.lib						
<ul> <li>Create Executable: Exp-DAC-DMA-TIMER_CM4\Exp-DAC-DMA</li> <li>✓ Debug Information</li> <li>✓ Create HEX File</li> <li>✓ Browse Information</li> <li>C create Library: Exp-DAC-DMA-TIMER_CM4\Ex</li></ul>	ER_CM4.lib						

W Options for Target 'Exp-DAC-DMA-TIMER_CM4'	×
Device Target Output Listing User C/C++ Asm Linker Debug Utilities	1
Select Folder for Listings Page Width: 79 - Page Length: 66 -	
Assembler Listing:\\\*Jst Cross Reference	
Compiler Listing:	]
C Preprocessor Listing:\\\\*i	
✓ Linker Listing:\\\\\Exp-DAC-DMA-TIMER_CM4.map	1
☑         Memory Map         ☑         Symbols         ☑         Size Info	
✓ Callgraph     ✓ Cross Reference     ✓ Totals Info	
✓ Unuse <u>a</u> Sections into	
OK Cancel Defaults Help	
Options for Target 'Exp-DAC-DMA-TIMER_CM4'	×
Device Target Output Listing User C/C++ Asm Linker Debug Utilities	
Preprocessor Symbols	
Define: CORE_CM4.USE_HAL_DRIVER.STM32H755xx	
Execute-only Code	
Optimization: Level 3 (-O3)   Enum Container always int	hum <u>b</u> Mode
Optimize for Time  Plain Char is Signed  No	o Auto Includes
□ Split Load and Store Multiple □ Read-Only Position Independent ▼ C	99 Mode
✓         One <u>E</u> LF Section per Function         □ <u>R</u> ead-Write Position Independent         □         Gi	NU extensions
Include/CM4/Inc;/Drivers/STM32H7xx_HAL_Driver/Inc;/Drivers/STM32H7xx_HAL_Driver/	Inc/Legacy
Misc Controls	
Compiler control string -c99 -ccpu Cortex-M4.fp.sp -DEVAL -DMICROLIB -g -O3apcs=interworksplit_se	ections -I 🔨
OK Cancel Defaults	Help

Options for Target 'Exp-DAC-DMA-TIMER_CM4'	×
Device Target Output Listing User C/C++ Asm Linker Debug Utilities	
Use Memory Layout from Target Dialog       X/O Base:         Make RW Sections Position Independent       R/O Base:         Make RO Sections Position Independent       R/O Base:         Don't Search Standard Libraries       Misable Warnings:	
Scatter File stm32h755xx_flash_CM4.sct 🗨 Edit.	-
Linker control string	~
OK Cancel Defaults	Help
c. Compilez avec F7 et vous devriez obtenir ceci : compiling stm32h7xx_hal_pwr_ex.c compiling stm32h7xx_hal_uart_ex.c compiling stm32h7xx_hal_tim_ex.c compiling system_stm32h7xx_dualcore_boot_cm4_cm7.c compiling stm32h7xx_hal_tim.c compiling stm32h7xx_hal_uart.c	

Build Time Elapsed: 00:02:20 d. Choisir maintenant la cible CM7

linking...

e. Ouvrez les options du projet avec ALT+F7 et vérifiez scrupuleusement les écrans suivants :

FromELF: creating hex file...
"Exp-DAC-DMA-TIMER\_CM4.axf" - 0 Error(s), 1 Warning(s).

Program Size: Code=1200 RO-data=712 RW-data=20 ZI-data=7332

<b>W</b> Options for	r Target 'Exp-DAC-	DMA-TIMER_C	M7'					×
Device Targe	et Output Listing	User C/C++	Asm	Linker	Debug U	Itilities		
STMicroelectro	onics STM32H755ZI	Tx:CM7		Code G	eneration			
		Xtal (MHz): 64.	0	ARM	Compiler:	Use defaul	t compiler versio	on 5 💌
Operating syst	tem: None		-					
System Viewe	r File:			I Us	se Cross-N se MicroLl	lodule Optimiz B	ation Big Endian	
STM32H755	_CM7.svd			Floatin	ng Point H	ardware: Do	uble Precision	•
Use Cust	om File							
Read/Only	Memory Areas			-Read/\	Nrite Mem	ory Areas		
default off-	chip Start	Size	Startup	default	off-chip	Start	Size	Nolnit
RO RO	DM1:		0		RAM1:			
□ RO	)M2:		0		RAM2:			
□ R0	)M3:		0		RAM3:			
on	-chip				on-chip	0.00000000	0.00000	
	M1: 0x8000000	0x100000			IRAM1:	0x20000000	0x20000	
L IRO	)M2:				IRAM2:	0x24000000	0×80000	
		ОК	Ca	ncel	Defau	ults		Help
Options for	or Target 'Exp-DAC		`M7'					~
			. ] .	1	- 1	1		~
Device   Targ	et   Output   Listing	User C/C+	+ Asm	Linker	Debug   l	Jtilities		
- Preproces	sor Symbols							
<u>D</u> efine:	CORE_CM7,USE_	HAL_DRIVER,S	TM32H75	5кх				
Undefine:								
	/ Code Generation -							
Execut	te-only Code		Strict AN	SIC		<u>W</u> aming	s: All Warnings	•
<u>O</u> ptimizatio	n: Level 3 (-03)	-	Enum <u>C</u> o	ntainer alw	ays int		Thum <u>b</u> M	lode
i Optimiz	ze <u>f</u> or Time		Plain Cha	ar is Signed			No Auto	Includes
	bad and Store Multipl	e 🗆	Read-On	ly Position	Independe	ent	C99 Mod	e
I One E	LF Section per Funct	ion I	Read-Wr	ite Position	Independ	ent	GNU exte	ensions
Include Patha	/CM7/Inc;/Drive	ers/STM32H7xx_	_HAL_Driv	er/Inc;/D	rivers/STN	132H7xx_HAL	_Driver/Inc/Leg	jacy
Misc								
Controls		117( 1 5		MICDOLU			ba ar	
control	/CM7/Inc -I/Dri	vers/STM32H7x	EVAL -D_ x_HAL_Dr	_MICROLIE iver/Inc -I	3 -g -03a	apcs=interwork	split_sections	
string	1							v
		01/	1 0	1	D (			Usta
		UK	L Ca	ncei	Deta	uits		Help

-								
W Options fo	or Target 'Exp-DAC-DMA-TIMER_CM7			×				
Device Tar	et Output Listing User C/C++	Asm Linker Debug	Utilities					
Use Mer	mory Layout from Target Dialog RW Sections Position Independent RO Sections Position Independent & Search Standard Libraries ort 'might fail' Conditions as Errors	<u>X</u> /O Base: <u>R</u> /O Base: R/ <u>W</u> Base <u>d</u> isable Warnings:	0×08000000 0×20000000					
Scatter File	stm32h755xx_flash_CM7.sct		▼	E dit				
controls Linker control string	Linker      cpu Cortex-M7.fp.dp *.o         control       -library_type=microlibstrictscatter "Exp-DAC-DMA-TIMER_CM7\Exp-DAC-DMA-TIMER_CM7.sct"							
	ОК	Cancel Defa	aults	Help				
Compilez avec compiling stm compiling sys compiling stm compiling stm linking Program Size: "Exp-DAC-DMA- Build Time El	c F7 et vous devriez obtenir ceci 32h7xx_hal_pcd.c tem_stm32h7xx_dualcore_boot_cm4_ 32h7xx_ll_usb.c 32h7xx_hal_uart.c Code=20344 RO-data=720 RW-data= TIMER_CM7\Exp-DAC-DMA-TIMER_CM7. apsed: 00:03:08	_cm7.c =20 ZI-data=268704204 .axf" - 0 Error(s), 1	H Warning(s).					

## 4 CONCEPTION ET CODAGE

## 4.1 CONCEPTION

f.

Nous souhaitons créer une forme d'onde sinusoïdale.

- 1) Ouvrir la note d'application fournie par ST : AN3126.
- 2) En 2.1.1 se trouve l'explication pas à pas de la méthode à suivre pour créer une forme sinusoidale.

#### 2.1.2 Waveform preparation

To prepare the digital pattern of the waveform, we have to go through some mathematics.

Our objective is to have ten digital pattern data (samples) of a sine wave form that varies from 0 to  $2\pi$ .





The sampling step is  $2\pi / n_s$  (number of samples).

The value of sin(x) varies between -1 and 1, we have to shift it up to have a positive sine wave with samples varying between 0 and 0xFFF (corresponding to the 0 to 3.3 V voltage range, where  $V_{RFF}$  is set to 3.3 V).

$$y_{\text{SineDigital}}(x) = \left(\sin\left(x \cdot \frac{2\pi}{n_s}\right) + 1\right)\left(\frac{(0xFFF+1)}{2}\right)$$

3) Page 16 on retrouve la formule permettant de calculer la sortie du DAC

Digital inputs are converted to output voltages on a linear conversion between 0 and  $V_{REF+}$ . The analog output voltages on each DAC channel pin are determined by the equation

$$DAC_{Output} = V_{REF} \frac{DOR}{DAC_MaxDigital Value + 1}$$

**Vref** = 3,3 V

**DOR** est le registre dans lequel on met la valeur relative du signal à convertir en tension. **DAC\_MaxDigitalValue** dépend de la résolution que l'on utilise (12 bits ou 8 bits)

#### 4.2 PROGRAMMATION

Ouvrir le fichier main.c du CM7 et réaliser ce qui est indiqué ci-dessous.

Inclure «math.h »:

```
31 /* Private includes -----*/
32 /* USER CODE BEGIN Includes */
33 #include "math.h" // DAC1 : pour utilisation de la fonction sinus
34 /* USER CODE END Includes */
```

Ajouter un define pour la constante PI :

```
41 /* Private define -----*/
42 /* USER CODE BEGIN PD */
43 #define PI 3.1415926 // DAC1 : pour utilisation de la fonction décrite dans AN3126
44 #define HSEM_ID_0 (OU) /* HW semaphore 0*/
45 /* USER CODE END PD */
```

Nous allons avoir besoin de quelques variables et là, il ne faut pas louper l'attribut qui sert à indiquer au linker où doit se placer le tableau dont nous aurons besoin !!! Ce n'est dit nulle-part mais le DMA associé au DAC ne fonctionne pas si l'on ne place pas la zone mémoire en 0x30000000 !!!

```
54 /* USER CODE BEGIN PV */
55 uint32_t sine_val[100] __attribute__((at(0x3000000)));
56 int n = 100; // DAC1 : Nb échantillons
57 int i;
58 /* USER CODE END PV */
```

Il faut maintenant créer un tableau de 100 échantillons. Puisque nous aurons une interruption dont la fréquence sera de 10 kHz, chaque échantillon sera produit toutes les 100  $\mu$ s et comme nous en aurons 100 par forme d'onde, la période de l'onde générée sera de 100\*100 $\mu$ s = 10 ms, c'est dire que la fréquence de l'onde générée sera de 1/0,01 = **100 Hz**.

Ce que confirme d'ailleurs la note d'application :

#### 2.1.3 Setting the sine wave frequency

To set the frequency of the sine wave signal, the user has to set the frequency ( $f_{TimerTRGO}$ ) of the timer trigger output.

The frequency of the produced sine wave is

$$f_{\text{Sinewave}} = \frac{f_{\text{TimerTRGO}}}{n_{\text{s}}}$$

Nous aurons besoin de la constante PI = 3,1415926.

Nous créerons une fonction **get\_sineval()** dont le but sera de remplir le tableau d'échantillons avec les valeurs définies par la formule de la note d'application AN3126 (cf plus haut).

$$y_{\text{SineDigital}}(x) = \left(\sin\left(x \cdot \frac{2\pi}{n_s}\right) + 1\right)\left(\frac{(0xFFF+1)}{2}\right)$$

Dans notre cas, le nombre de samples **ns** sera de 100, et 0xFFF correspond à la résolution du DAC sur 12 bits. 0xFFF+1 = 4096 en décimal et donc (0xFFF+1)/2=2048.

Dans la fonction main() il faudra

- démarrer le timer tim2 en appelant l'API HAL\_TIM\_Base\_Start(&htim2) ;
- appeler la fonction get\_sineval() qui servira une seule fois pour remplir la zone mémoire avec les 100 valeurs de samples.
- démarrer le DAC avec l'API spéciale liée au DMA :
   HAL\_DAC\_Start\_DMA(&hdac1,DAC1\_CHANNEL\_1,sine\_val,100,DAC\_ALIGN\_12B\_R);

Dans cette API, on passe :

- en 3<sup>ème</sup> paramètre, le pointeur sur la zone où sont stockés les échantillons ;
- en 4<sup>ème</sup> paramètre, le nombre d'échantillons.

Il est temps de définir la fonction servant à remplir le tableau d'échantillons.

```
66  /* Private user code ------*/
67  /* USER CODE BEGIN 0 */
68  void get_sineval() // DACl : usage unique pour valorisation des samples ; n = nombre d'échantillons
69  [// Cf formule note d'application AN3126
70  for(int i=0;i<n;i++)
71  {
72  sine_val[i]=((sin(i*2*PI/n)+1)*2048);
73  }
74  ]
75  /* USER CODE END 0 */</pre>
```

Et il faut maintenant réaliser la première initialisation du tableau, puis démarrer le timer et enfin le DMA associé au DAC1.

Il reste à :

- compiler le main du CM7 via la touche F7 ;

```
Build Output

*** Using Compiler 'V5.06 update 6 (build 750)', folder: 'D:\Programmes\Keil_v5\ARM\ARMCC\Bin'

Build target 'Exp-DAC-DMA-TIMER_CM7'

compiling main.c...

linking...

Program Size: Code=23744 RO-data=1008 RW-data=28 ZI-data=268704196

"Exp-DAC-DMA-TIMER_CM7\Exp-DAC-DMA-TIMER_CM7.axf" - 0 Error(s), 0 Warning(s).

Build Time Elapsed: 00:00:12
```

- charger le programme CM7 dans le STM32H755ZI via la touche F8 ;

```
Build started: Project: Exp-DAC-DMA-TIMER
*** Using Compiler 'V5.06 update 6 (build 750)', folder: 'D:\Programmes\Keil_v5\ARM\ARMCC\Bin'
Build target 'Exp-DAC-DMA-TIMER_CM7'
compiling main.c...
linking...
Program Size: Code=23744 RO-data=1008 RW-data=28 ZI-data=268704196
"Exp-DAC-DMA-TIMER_CM7\Exp-DAC-DMA-TIMER_CM7.axf" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:12
Load "Exp-DAC-DMA-TIMER_CM7\LExp-DAC-DMA-TIMER_CM7.axf"
**** error 129: MapMem - map size truncated to 128MB
Erase Done.
Programming Done.
Verify OK.
Flash Load finished at 11:15:18
```

- De nouveau sélectionner la cible CM4 comme réalisé au chapitre 3, opération 15) a.
- Tenter de charger le programme CM7 dans le STM32H755ZI via la touche F8...

Cannot Load Flash Device Description!		
	ption!	Cannot Load Flash Device Description
D:\Documents\Informatique\Programmes\Arm\Packs\Keil\STM 32H7xx_DFP\2.5.0\Flash\STM32H7xx.FLM	grammes\Arm\Packs\Keil\STM 7xx.FLM	D:\Documents\Informatique\Progra 32H7xx_DFP\2.5.0\Flash\STM32H7xx



Selon l'état de développement de STM32CubeMX au moment où vous exécuterez ce tutoriel, vous aurez ou pas les messages d'erreur ci-dessus...

Pour en venir à bout, voici ce qu'il faut faire :

iqi	ue\Prog	rammes\S	STM32\NUCL	EO-H	755Z
t	Flash	Debug	Peripherals	Тос	ols !
	LOAD D	ownload		F8	19
ų	E	rase			
	c	onfigure F	lash Tools		LX
		n			e

2) Appuyer sur Settings

vice   Target	Output   Listing   User   C/C++	Asm Linker Debug Utilities
Configure Flash	Menu Command	
• Use Target	Driver for Flash Programming	Vse Debug Driver
	Use Debug Driver	Settings Update Target before Debugging
Init File:		Edit
Arguments:	Run Independent	
Output File:	FILE Processing (FCARM):	Add Output File to Group:
		Application/MDK-ARM
Image Files Ro	ot Folder:	Generate Listing

## 3) Appuyer sur Add

Cortex-M Target Driver Setup

Debug Trace Flash Download Download Function C Erase Full Chip C Erase Sectors C Do not Erase	Pack Program	RAM for / Start: 0	Algorithm x20000000	Size: 0x000010		
Programming Algorithm						
Description	Device Size	Device Type	Address	s Range		
	Add	Start:		Size:		
				ОК А	vnuler	Appliquer
Choisir STM32H7x_2048 e	t appuyer s	ur ADD				
Add Flash Programming Algorit	hm			×		

 $\times \mid$ 

Description	Flash Size	Device Type	Origin	^
MT25TL01G_STM32H745I	128M	Ext. Flash 16-bit	Device Family Package	
MT25TL01G_STM32H747I	128M	Ext. Flash 16-bit	Device Family Package	
MT25TL01G_STM32H750B	128M	Ext. Flash 16-bit	Device Family Package	
MTFC4GACAJCN_STM32H	512M	Ext. Flash 16-bit	Device Family Package	
MX25LM51245G_STM32H7	64M	Ext. Flash SPI	Device Family Package	
MX25LM51245G_STM32H7	64M	Ext. Flash SPI	Device Family Package	
MX25LM51245G_STM32H7	64M	Ext. Flash SPI	Device Family Package	
MX25LM51245G_STM32H7	64M	Ext. Flash SPI	Device Family Package	
M29W128GL_STM32H743I	16M	Ext. Flash 16-bit	Device Family Package	
M29W128GL_STM32H7B3I	64M	Ext. Flash 16-bit	Device Family Package	
STM32H7x_2048	2M	On-chip Flash	Device Family Package	
STM32H7xx_MT25TL01G	64M	Ext. Flash SPI	Device Family Package	
STM32H7xx_MT25TL01G	128M	Ext. Flash SPI	Device Family Package	
AM29x128 Flash	16M	Ext. Flash 16-bit	MDK Core	
K8P5615UQA Dual Flash	64M	Ext. Flash 32-bit	MDK Core	
I PC18vv/43vv MX25\/8035F	8M	Evt Flash SPI	MDK Core	~
Selected Flash Algorithm File:				
D:\Documents\Informatique\Pro	grammes\Arm\	Packs\Keil\STM32	H7xx_DFP\2.5.0\CMSIS\Flash	\STM32

5) Mettre 0x10000000 dans start et 0x00001000 dans size et appuyer sur OK

Download Function	I   Pack   P ▼ Program	RAM for A	lgorithm	
C Do not Erase	Reset and R	un		
Programming Algorithm		D		
Description STM32H7x 2048	Device Size 2M	On-chip Flash	Address Range 08000000H - 081FFFFFH	
		Start: 0	x08000000 Size: 0x00200000	
	Add	Remove		
	Add	Remove		
	Add	Remove		

6) 7) Appuyer encore sur OK dans la fenêtre mère.

8) Appuyer de nouveau sur F8 pour charger le programme du CM4

Si malgré ceci vous avez encore des messages d'erreur, revérifiez toutes les options comme indiqué au chapitre 3, opération 15) !!! Celle-ci est particulièrement délicate, pour CM4 il ne faut pas cocher la case, mais pour CM7 oui :

Options fo	r Target 'Exp-DAC-DN	MA-TIMER_CM4'			×
Device Targ	et Output Listing U	lser C/C++ Asm	Linker Debug	Utilities	
Use Men Make Dont Repo	nory Layout from Target RW Sections Position RO Sections Position I Search Standard Librar ort 'might fail' Conditions a	Dialog Independent ndependent ies as Errors	X/O Base: R/O Base: R/W Base disable Warnings:	0x08000000 0x20000000	
Scatter File	stm32h755xx_flash_Cl	M4.sct		•	Edit
Misc controls					< >
Linker control string	cpu Cortex-M4.fp.sp -library_type=microlib	*.o -strictscatter ''stm32	h755xx_flash_CM4.sc	t"	~
		ОК	Cancel Def	aults	Help
Options fo	or Target 'Exp-DAC-DI	MA-TIMER_CM7'			×
Options fo	or Target 'Exp-DAC-D! et Output Listing U	MA-TIMER_CM7' Iser C/C++ Asm	Linker Debug	Utilities	×
Device Targ	or Target 'Exp-DAC-DI et Output Listing U	MA-TIMER_CM7' Jser   C/C++   Asm Dialog	Linker Debug	Utilities	×
Device Targ	or Target 'Exp-DAC-DI et Output Listing U nory Layout from Target RW Sections Position	MA-TIMER_CM7' Jser   C/C++   Asm Dialog	Linker Debug <u>X</u> /O Base: <u>R</u> /O Base:	Utilities	×
Options for Device Targ Use Mer Make	or Target 'Exp-DAC-DI et Output Listing U nory Layout from Target RW Sections Position e RO Sections Position I	MA-TIMER_CM7' Iser C/C++ Asm Dialog Independent independent	Linker Debug X/O Base: <u>R</u> /O Base: R/ <u>W</u> Base	Utilities	×
Options for Device Targ Use Mer Make Make Don Report Report	or Target 'Exp-DAC-D! et Output Listing U nory Layout from Target RW Sections Position e RO Sections Position I t Search Standard Librar ort 'might fail' Conditions	MA-TIMER_CM7' Iser C/C++ Asm Dialog Independent independent ies as Errors	Linker Debug <u>X</u> /O Base: <u>R</u> /O Base: R/ <u>W</u> Base <u>d</u> isable Warnings:	Utilities	×
Coptions for Device Targ	et Output Listing U nory Layout from Target RW Sections Position RO Sections Position I t Search Standard Librar ort 'might fail' Conditions	MA-TIMER_CM7'	Linker Debug X/O Base: <u>R</u> /O Base: R/ <u>W</u> Base disable Warnings:	Utilities	Edit
Controls	er Target 'Exp-DAC-DI et Output Listing U nory Layout from Target RW Sections Position e RO Sections Position I t Search Standard Librar ort 'might fail' Conditions stm32h755xx_flash_Ci	MA-TIMER_CM7' Iser C/C++ Asm Dialog Independent ndependent ries as Errors	Linker Debug X/O Base: R/O Base: R/W Base disable Warnings:	Utilities	E dit
Controls	et Output Listing U nory Layout from Target RW Sections Position RO Sections Position t Search Standard Librar ort 'might fail' Conditions stm32h755xx_flash_C	MA-TIMER_CM7' Iser C/C++ Asm Dialog Independent nes as Errors M7.sct *.0 *.0	Linker Debug X/O Base: R/O Base: R/ <u>W</u> Base disable Warnings:	Utilities	Edit

Enfin, vous n'êtes pas dispensé de regarder ce qui se passe dans le fichier .map que je vous laisse le soin de découvrir, il vous informera notamment du placement définitif du tableau sine\_val dans le bon segment de mémoire 0x30000000 !

Pour ce premier test, Il n'y a rien du tout à écrire dans la boucle infinie, puisque c'est la fonction DMA qui se charge de tout envoyer au DAC !

Le processeur n'a donc aucun autre traitement à exécuter que la boucle infinie.

Arrivé à ce point du tutoriel, si vous branchez un oscilloscope (masse su CN8 position 11 et signal PA4 sur CN7 position 17) alors vous devriez observer une belle sinusoïde de 100 Hz.

#### NUCLEO-H745ZI-Q and NUCLEO-H755ZI-Q extension connectors



### 4.3 PETITE EVOLUTION

On voudrait bien voir évoluer un peu la fréquence de cette sinusoïde. Il y a plusieurs façon de le faire, en modifiant les paramètres du timer ou en redéfinissant le nombre d'échantillons de la courbe...

J'ai tellement eu de mal à faire fonctionner ce programme avec une tableau de variables, que c'est la deuxième solution que je vais choisir. Toutes les 3 secondes, je changerai le nombre d'échantillons pour faire passer la fréquence de  $100 \rightarrow 133 \rightarrow 200 \rightarrow 400 \rightarrow 100$  etc.

Rien de plus simple, il suffit que faire évoluer n cycliquement avec les valeurs suivantes : 100, 75, 50, 25, 100, 75... Ce qui donne le code suivant :

```
146
       /* Infinite loop */
147
        /* USER CODE BEGIN WHILE */
148
        while (1)
149 🛱 - {
         /* USER CODE END WHILE */
150
151
152 /* USER CODE BEGIN 3 */
      HAL_Delay(3000);
HAL_DAC_Stop_DMA(&hdacl,DAC1_CHANNEL_1);
153
154
         if (n==25) n=125 ;
155
         n = n - 25;
156
         get_sineval();
HAL_DAC_Start_DMA(&hdacl,DACl_CHANNEL_l,sine_val,n,DAC_ALIGN_12B_R);
157
158 HAL_DAC_Start_DMA(&)
159 - }
160 /* USER CODE END 3 */
```