

PROJEKT

STEROWNIKI ROBOTÓW

Założenia projektowe

Theremin THR

Skład grupy:

Cyprian HRYNIUK, 235512

Tomasz MASŁOŃ, 235827

Termin: srTN17

Prowadzący:

mgr inż. Wojciech DOMSKI

13 marca 2019

Spis treści

1	Opis projektu	2
2	Konfiguracja mikrokontrolera	3
2.1	Konfiguracja pinów	5
2.2	I2C	5
2.3	UART	5
2.4	DMA	5
3	Harmonogram pracy	6
3.1	Podział pracy	6
4	Podsumowanie	6
	Bibilografia	7

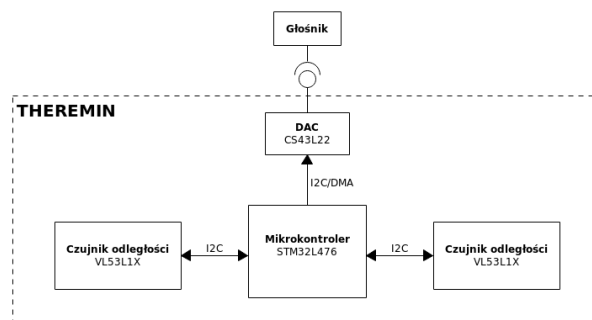
1 Opis projektu

Celem projektu jest stworzenie theremina opartego na płytce rozwojowej STM32L476G-DISCO [2] i dwóch cyfrowych czujnikach odległości VL53L1X [4] korzystających z technologii Time-of-Flight. Dane z jednego czujnika będą wpływać na częstotliwość dźwięku a dane z drugiego na głośność. Aby uzyskać zamierzony efekt niezbędne będzie zaprogramowanie obsługi czujników oraz Audio DAC [1] zawartego na płytce.

Głównym zadaniem jest napisanie pełnego oprogramowania na MCU STM32L476VGT6 [3] pozwalającego na generowanie fali dźwiękowej (sinusoidea). Modyfikacje do generowanej fali będą wprowadzane ruchami rąk (bliżej-dalej). Informacja na temat odległości będzie podawana przez czujniki VL53L1X. Na początku zostanie zaimplementowana funkcja generująca podstawową sinusoidę, funkcję tę później rozwinimy tak, aby generowała ciekawe brzmienie.

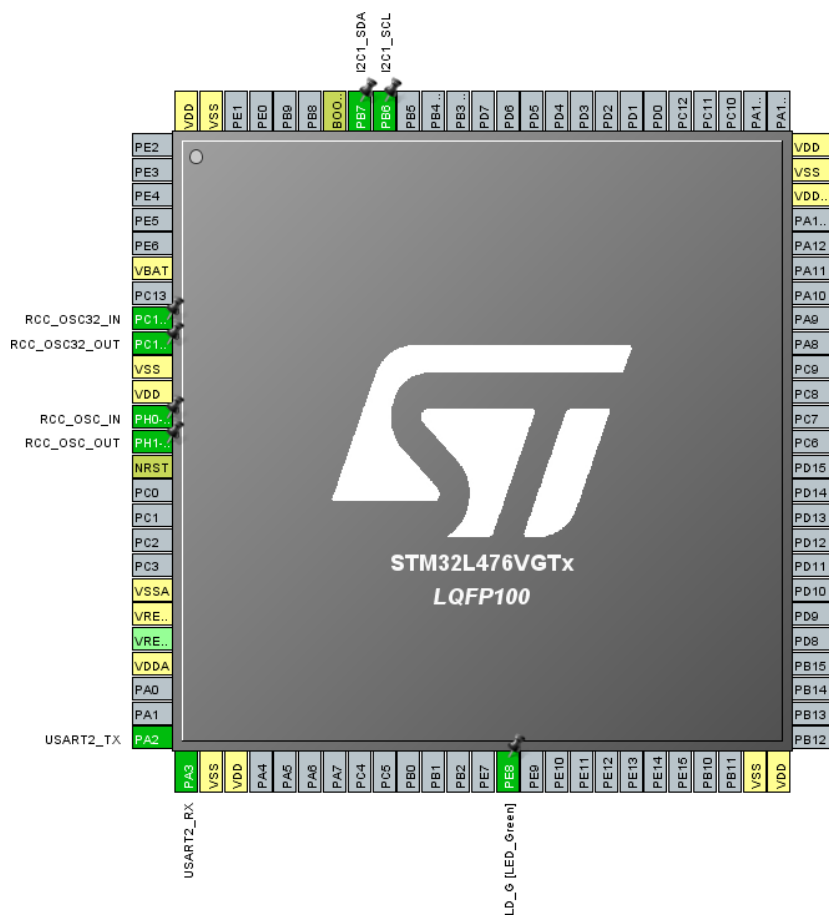
Czujniki będą się komunikować z MCU za pomocą protokołu I^2C . Ustawienia Audio DAC również będą realizowane za pomocą protokołu I^2C . Przekazanie próbek do Audio DAC odbędzie się za pomocą DMA.

Projekt będzie tworzony w języku C korzystając ze środowiska Atollic TrueSTUDIO for STM32. Konfiguracja peryferiów zostanie zrealizowana przy pomocy programu STM32CubeMX. Wszelkie dokumenty dotyczące projektu będą powstawać w systemie składu tekstu \LaTeX . Na potrzeby projektu zostało założone zdalne repozytorium w hostingowanym serwisie internetowym *github.com* korzystającym z systemu kontroli wersji *git*. Repozytorium jest publiczne i dostępne pod linkiem: <https://github.com/CypHry/Theremin>



Rysunek 1: Architektura systemu

2 Konfiguracja mikrokontrolera



2.1 Konfiguracja pinów

Numer pinu	PIN	Tryb pracy	Funkcja/etykieta
8	PC14	OSC32_IN	RCC_OSC32_IN
9	PC15	OSC32_OUT	RCC_OSC32_OUT
12	PH0	OSC_IN	RCC_OSC_IN
13	PH1	OSC_OUT	RCC_OSC_OUT
39	PE8	GPIO_Output	LD_G [LED_Green]
47	PB10	I2C2_SCL	
48	PB11	I2C2_SDA	
86	PD5	USART2_TX	USART_TX
87	PD6	USART2_RX	USART_RX
92	PB6	I2C1_SCL	
93	PB7	I2C1_SDA	

Tabela 1: Konfiguracja pinów mikrokontrolera

2.2 I2C

Parametr	Wartość
I2C Speed Mode	Standard Mode
I2C Speed Frequency	100kHz

Tabela 2: Konfiguracja peryferium I2C

2.3 UART

Peryferium skonfigurowano na potrzeby debugowania.

Parametr	Wartość
Baud Rate	9600
Word Length	8 Bits (including parity)
Parity	None
Stop Bits	1
Data Direction	Receive and Transmit

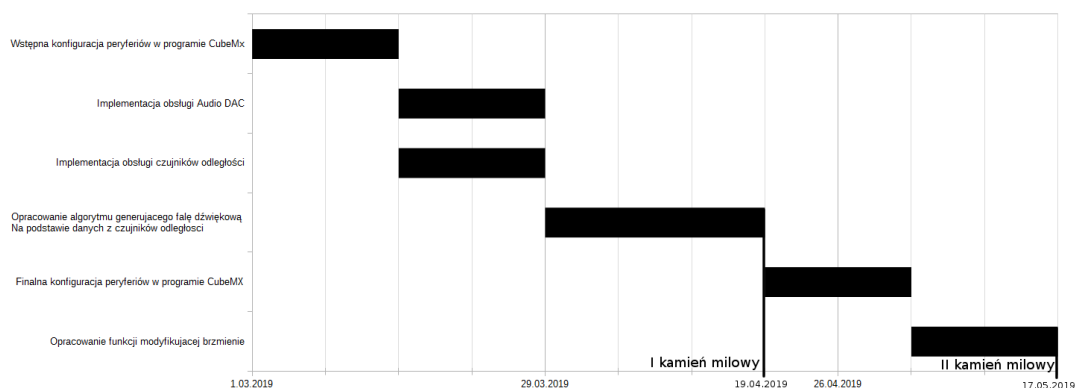
Tabela 3: Konfiguracja peryferium USART

2.4 DMA

Parametr	Wartość
DMA request	DAC_CH1
Stream	DMA1_Channel3
Direction	Memory To Peripheral

Tabela 4: Konfiguracja peryferium DMA

3 Harmonogram pracy



Rysunek 4: Diagram Gantta

Kolejne kamienie milowe pokrywają się z etapami wyszczególnionymi w podrozdziale Podział pracy.

3.1 Podział pracy

Tomasz Masłoń	%	Cyprian Hryniuk	%
Wstępna konfiguracja peryferiów w programie CubeMx		Wstępna konfiguracja peryferiów w programie CubeMx	
Implementacja obsługi Audio DAC		Implementacja obsługi czujników odległości	
Opracowanie algorytmu generującego falę dźwiękową na podstawie danych z czujników odległości		Opracowanie algorytmu generującego falę dźwiękową na podstawie danych z czujników odległości	

Tabela 5: Podział pracy – Etap II

Tomasz Masłoń	%	Cyprian Hryniuk	%
Finalna konfiguracja peryferiów w programie CubeMX		Finalna konfiguracja peryferiów w programie CubeMX	
Opracowanie funkcji modyfikującej brzmienie		Opracowanie funkcji modyfikującej brzmienie	

Tabela 6: Podział pracy – Etap III

4 Podsumowanie

Projekt dotyczy stworzenia instrumentu muzycznego - theremina, przy wykorzystaniu cyfrowych czujników odległościowych oraz AudioDAC

Literatura

- [1] CirrusLogic. CS43L22 Datasheet. 2010.
- [2] STMicroelectronics. UM1879 User manual Discovery kit with STM32L476VG MCU. Maj 2017.
- [3] STMicroelectronics. STM32L476xx Datasheet. Maj 2018.
- [4] STMicroelectronics. VL53L1X Datasheet. 2018.