**Laboratorium**

Systemy wbudowane

**Temat projektu**

Układ sterujący stanem 8 wyjść za pomocą komend wysyłanych z terminala

Daniel Dunak

Michał Kuśka

19.01.2022r

1. **Opis projektu**
   1. **Wykorzystane elementy sprzętowe**

Dioda LEDx8, Konwerter poziomów RS232, stm32f407 discovery

* 1. **Wymagania:**

**1.** Urządzenie jest podłączone do komputera PC na którym pracuje program terminala

**2.** Urządzenie odbiera szeregowo dane z PC. Każda komenda jest zakończona znakiem LF.

Komendy:

ONi – (gdzie i należy do przedziału<0-7> jest numerem wyjścia). Komenda powoduje włączenie wyjścia nr i.

OFFi –Wyłączenie wyjścia nr i.

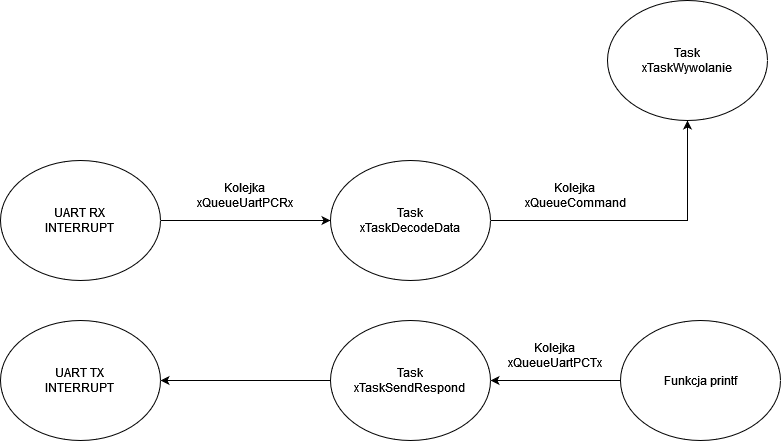
ONALL – Włącz wszystkie wyjścia.

OFFALL – Wyłącz wszystkie wyjścia.

AT – Attention. Przesyła do terminala informację (np.. „Nasz wspanialy projekt v 2012”)

**3.** Każda odebrana komenda powinna być potwierdzone poprzez przesłanie do PC albo ciągu liter „OK.”. jeśli komenda była poprawna albo „ERR” w przypadku gdy komenda nie została rozpoznana.

1. **Opis wykonanego projektu**



Rysunek Diagram działania programu

Tabela Opis wyjść z mikrokontrolera

|  |  |
| --- | --- |
| Numer pinu | Rola |
| PC1 | Dioda nr.0 |
| PC3 | Dioda nr.1 |
| PA1 | Dioda nr.2 |
| PA3 | Dioda nr.3 |
| PA5 | Dioda nr.4 |
| PA7 | Dioda nr.5 |
| PC5 | Dioda nr.6 |
| PB1 | Dioda nr.7 |
| PC12 | UART5\_TX |
| PD2 | UART5\_RX |

Tabela Szczegółowy opis zawartości poszczególnych kolejek

|  |  |  |
| --- | --- | --- |
| Komenda (PC->dekoder) | Komenda (dekoder->GPIO) | Cel |
| ON0 | 0x10 | Zapalenie diody nr.0 |
| ON1 | 0x11 | Zapalenie diody nr.1 |
| ON2 | 0x12 | Zapalenie diody nr.2 |
| ON3 | 0x13 | Zapalenie diody nr.3 |
| ON4 | 0x14 | Zapalenie diody nr.4 |
| ON5 | 0x15 | Zapalenie diody nr.5 |
| ON6 | 0x16 | Zapalenie diody nr.6 |
| ON7 | 0x17 | Zapalenie diody nr.7 |
| OFF0 | 0x20 | Wyłączenie diody nr.0 |
| OFF1 | 0x21 | Wyłączenie diody nr.1 |
| OFF2 | 0x22 | Wyłączenie diody nr.2 |
| OFF3 | 0x23 | Wyłączenie diody nr.3 |
| OFF4 | 0x24 | Wyłączenie diody nr.4 |
| OFF5 | 0x25 | Wyłączenie diody nr.5 |
| OFF6 | 0x26 | Wyłączenie diody nr.6 |
| OFF7 | 0x27 | Wyłączenie diody nr.7 |
| ONALL | 0x30 | Zapalenie wszystkich diod |
| OFFALL | 0x40 | Wyłączenie wszystkich diod |

Za pomocą przerwania wypełniana jest kolejka xQueueUartPCRx zawiera ona znaki ASCI. Zadanie xTaskDecodeData na bieżąco dekoduje odebrane bajty. Wystąpienie znaku LF w odebranych kodach powoduje zakończenie dekodowania. Na tym etapie zadanie sprawdza, czy wykryto jakąś ze znanych komend. Kazda komenda ma swój idywidualny numer która zostanie przekazany, jeśli zostanie wykryta odpowiednia komenda. Numer ten trafia do kolejki XQueueCommand. Zadanie xTaskWywolanie na bieżąco kontroluje stan portów GPIO. Za pomocą funkcji printf napełniana jest kolejka o nazwie XQueueUartPCTx. Zadanie XTaskSendRespond odpowiedzialne jest za wysyłanie odpowiedzi poprzez port UART przy użyciu przerwania.