

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра «Телекомунікації»

Методичні вказівки  
до виконання лабораторної роботи №4  
з дисципліни «Вбудовані системи»  
на тему «Робота з інтерфейсом UART»

Львів 2021

**Мета роботи:** ознайомитися із принципами організації взаємодії мікроконтролера з персональним комп'ютером. Вивчити функціонування і програмування інтерфейсу UART.

### Короткі теоретичні відомості

**USART** (Universal Synchronous and Asynchronous Serial Receiver - Transmitter) – універсальний синхронний/асинхронний послідовний приймач/передавач. USART – внутрішній периферійний модуль МК призначений для обміну інформацією з іншими пристроями за правилами стандарту **RS-232**. RS-232 (Recommended Standard 232) – в телекомунікаціях, стандарт послідовного синхронного й асинхронного передавання двійкових даних між двома пристроями. Найчастіше використовується в промисловому і вузькоспеціальному обладнанні, вбудованих пристроях. Присутній в застарілих ПК як COM-порт. Швидкості обміну, підтримувані портом вимірюються в Бодах (1 Бод = 1 біт/с): 1200; 2400; 4800; 9600; 14400; 19200; 28800; 38400; 57600; 76800; 115200. Для сучасних пристроїв, максимальна швидкість може досягати і більших значень.

Основні властивості модуля:

- повнодуплексний обмін даними;
- синхронний і асинхронний режими роботи;
- синхронізація як від ведучого, так і від веденого пристрою;
- швидкість передавання може варіюватися в досить великих межах;
- слово даних довжиною 5...9 розрядів (в асинхронному режимі 8...9 розрядів), найчастіше – 8;
- апаратна підтримка генерації і контролю біта парності для виявлення помилок при прийомі;
- три переривання: «передачу завершено», «регістр даних передавача – порожній», «прийом завершено».

У МК ATmega328 для роботи з модулем USART використовуються наступні регістри:

- 3 керуючі регістри **UCSR0A**, **UCSR0B**, **UCSR0C**;
- 2 регістри швидкості передавання **UBRR0L** і **UBRR0H**;
- регістр даних **UDR0**.

Регістр А управління і статусу USART – **UCSR0A**

Bit	7	6	5	4	3	2	1	0	
	<b>RXCn</b>	<b>TXCn</b>	<b>UDREN</b>	<b>FE<sub>n</sub></b>	<b>DOR<sub>n</sub></b>	<b>UP<sub>n</sub></b>	<b>U2X<sub>n</sub></b>	<b>MPCM<sub>n</sub></b>	<b>UCSRnA</b>
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

**RXCn** - прапор завершення прийому. Встановлюється в 1 за наявності непрочитаних даних в буфері приймача (регістр даних **UDR0**). Скидається апаратно після спустошення буфера.

**TXCn** - прапор завершення передачі. Встановлюється в 1 після передачі всіх розрядів посилки з зсувного регістра передавача, за умови, що в регістр даних **UDR0** не було завантажено нового значення. Скидається апаратно при виконанні підпрограми обробки переривання або програмно, записом в нього логічної 1.

**UDREN** - прапор звільнення регістра даних USART. Встановлюється в 1 при порожньому буфері передавача (після пересилання байта з регістра даних **UDR0** у регістр зсуву передавача).

Установлений прапорець означає, що в регістр даних можна завантажувати нове значення. Якщо розряд **UDRIEn** регістра **UCRn** (UCSR0B) встановлено, генерується запит на переривання «регістр даних – порожній». Скидається апаратно, при записі в регістр даних.

**FEn** – прапор помилки кадрування. Встановлюється в 1 при виявленні помилки кадрування, тобто, якщо перший стоп-біт прийнятої посилки дорівнює 0. Скидається при прийомі стоп-біта рівного 1.

**DORn** – прапор переповнення. Встановлюється в 1, якщо новий кадр буде поміщено у регістр зсуву приймача до того, як з регістра даних буде прочитане попереднє слово. Прапорець скидається при пересиланні прийнятих даних зі зсувного регістра приймача в буфер.

**PEn** – прапор помилки контролю парності. Встановлюється в 1, якщо в даних, що знаходяться в буфері приймача, виявлена помилка контролю парності. При відключеному контролі парності цей розряд постійно читається як 0.

**U2Xn** – біт подвоєння швидкості обміну. Якщо цей біт встановлений в 1, коефіцієнт ділення передподільника частоти контролера зменшується з 16 до 8, подвоюючи тим самим швидкість асинхронного обміну по послідовному каналу. У USART розряд **U2Xn** використовується тільки при асинхронному режимі роботи. У синхронному режимі він повинен бути скинутий.

**MPCM** – біт включення режиму мультипроцесорного обміну. Якщо він встановлений в 1, ведений мікроконтролер очікує на прийом кадру, що містить адресу. Кадри які не містять адреси пристрою, ігноруються.

#### Регістр **B** управління і статусу - **UCSRB**

Bit	7	6	5	4	3	2	1	0	
	<b>RXCIEn</b>	<b>TXCIEn</b>	<b>UDRIEn</b>	<b>RXENn</b>	<b>TXENn</b>	<b>UCSZn2</b>	<b>RXB8n</b>	<b>TXB8n</b>	<b>UCSRnB</b>
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

**RXCIEn** – Дозвіл переривання по завершенню прийому. Якщо в даний розряд записано 1, то при встановленні прапора **RXCn** регістра UCSRnA генерується переривання.

**TXCIEn** – Дозвіл переривання по завершенню передачі. Якщо в даний розряд записано 1, то при встановленні прапора **TXCn** регістра UCSRnA генерується відповідне переривання.

**UDRIEn** – Дозвіл переривання при очищенні регістра даних USART. Якщо в даний розряд записано 1, то при встановленні прапора **UDREN** в регістрі UCSRnA генерується переривання.

**RXENn** – Дозвіл прийому даних.

**TXENn** – Дозвіл передачі даних.

**UCSZn2** – Формат посилки. У модулях USART він використовується спільно з розрядами **UCSZn1** і **UCSZn0** регістра UCSRnC.

**RXB8n** – 8-й розряд отриманих даних, при використанні 9-розрядних посилок. Вміст цього розряду має бути зчитаний до читання регістра даних **UDRn**.

**TXB8n** – 8-й розряд даних для передачі, при використанні 9-розрядних посилок. Необхідне значення має бути занесено в цей розряд до завантаження байта даних в регістр **UDRn**.

## Регістр С керування і статусу USART – UCSRC

Bit	7	6	5	4	3	2	1	0	
	<b>UMSELn1</b>	<b>UMSELn0</b>	<b>UPMn1</b>	<b>UPMn0</b>	<b>USBSn</b>	<b>UCSZn1</b>	<b>UCSZn0</b>	<b>UCPOLn</b>	<b>UCSRnC</b>
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	1	1	0	

**UMSELn1, UMSELn0** – Вибір режиму роботи USART.

UMSELn1	UMSELn0	Режим
0	0	Асинхронний
0	1	Синхронний
1	0	(резерв)
1	1	Ведучий SPI

**UPMn1, UPMn0** – Функціонування схеми контролю та формування парності

UPMn1	UPMn0	Режим контролю парності
0	0	Перевірку парності відключено
0	1	(резерв)
1	0	Перевірка на парність
1	1	Перевірка на непарність

**USBSn** – Кількість стоп-бітів. (*USBSn* = 0 – 1 стоп-біт, *USBSn* = 1 – 2 стоп-біти)

**UCSZn1, UCSZn0** – Формат посилки (011 – 8-біт)

UCSZn2	UCSZn1	UCSZn0	Формат посилки (біт)
0	0	0	5
0	0	1	6
0	1	0	7
0	1	1	8
1	0	0	-
1	0	1	-
1	1	0	-
1	1	1	9

**UCPOLn** – Задає полярність синхронізації *XCKn* (використовується тільки в синхронному режимі).

*Примітка: n* – означає номер периферійного модуля USART. (Для ATmega328 *n* = 0)

## Ініціалізація USART

Перед початком сеансу зв'язку необхідно виконати ініціалізацію USART, яка зазвичай складається з установки швидкості зв'язку, задання формату посилки і дозволу роботи передавача та приймача. Якщо використовується управління зв'язком по перериваннях, то під час ініціалізації необхідно, щоб був скинутий прапор загального дозволу переривань (необхідно заборонити всі переривання). Якщо необхідно виконати повторну ініціалізацію USART, наприклад, для зміни швидкості зв'язку або формату посилки, то необхідно переконатися, аби під час ініціалізації передача була припинена. Прапор ***TXCn*** може використовуватися для перевірки завершення роботи передавача, а прапор ***RXCn*** - для перевірки відсутності в приймальному буфері нелічених даних. Зауважимо, що при використанні прапора ***TXCn*** він повинен скидатися програмно перед початком кожної передачі (перед записом в ***UDRn***).

Формули розрахунку «швидкості передачі» UART:

```
// Для U2X = 0
```

```
UBRRnH = (F_CPU / (16 * baudrate) - 1) >> 8;
```

```
UBRRnL = F_CPU / (16 * baudrate) - 1;
```

```
// Для U2X = 1
```

```
UBRRnH = (F_CPU / (8 * baudrate) - 1) >> 8;
```

```
UBRRnL = F_CPU / (8 * baudrate) - 1;
```

## Передача даних

Робота передавача USART дозволяється встановленням біта дозволу передачі (***TXENn***) в регістрі ***UCSRnB***. Після чого альтернативна функція виводу (PD0) ***TXD*** із звичайного порта замінюється на функцію виходу послідовної передачі даних. Швидкість зв'язку, режим роботи і формат посилки мають встановлюватися одноразово перед початком передачі. Початок передачі зніціюється записом передаваних даних в буфер передавача. Буферизовані дані в буфері передавача будуть переміщені в регістр зсуву після того як він буде готовий до відправки нової посилки. Запис в регістр зсуву нових даних відбувається в стані чекання (коли передача завершена) або відразу після завершення передачі останнього стоп-біта попередньої посилки. Якщо в регістр зсуву записані нові дані, то починається передача однієї посилки на швидкості, визначеній в регістрі швидкості зв'язку.

Передавач USART має два прапори, які відображають його стан: прапор звільнення регістра даних ***UDREN*** і прапор завершення передачі ***TXCn*** в регістрі ***UCSRnA***. Обидва прапори можуть використовуватися для генерації переривань. Прапор звільнення регістра даних ***UDREN*** відображає готовність буфера передавача прийняти нові дані. Даний біт встановлюється при звільненні передавального буфера, і скидається коли буфер передавача містить дані для передачі. Прапор завершення передачі ***TXCn*** набуває значення 1, якщо вся посилка в регістрі зсуву передавача була повністю “зсунута” (передана) і в буфері передавача відсутні нові дані для передачі. Прапор ***TXCn*** автоматично скидається при переході на вектор обробки переривання по завершенню передачі або програмно скидається шляхом запису в нього логічної 1.

## Прийом даних

Робота приймача USART дозволяється записом логічної 1 в біт ***RXENn*** регістра **UCSRnB**. Після дозволу роботи приймача звичайне призначення виводу (PD0) **RXD** замінюється на вхід послідовного введення даних приймача USART. Швидкість зв'язку, режим роботи і формат посилки мають бути встановлені одноразово перед початком виконання прийому даних. Приймач починає прийом даних лише після визначення дійсного старт-біта. Вибірка наступних за старт-бітом біт-даних відбувається з частотою, рівною швидкості зв'язку. Після отримання першого стоп-біта, тобто коли послідовна посилка повністю прийнята і знаходиться в регістрі зсуву приймача, вміст регістра зсуву переміщується в буфер прийому. Буфер прийому зчитується при читанні регістра **UDRn**.

Прапор завершення прийому **RXC** у регістрі **UCSRnA** сигналізує про наявність непрочитаних даних у буфері прийому. Даний прапор дорівнює 1, якщо є непрочитані дані, і рівний 0, якщо буфер приймача вільний (тобто не містить яких-небудь даних). Якщо приймач відключається (***RXENn***=0), то приймальний буфер буде скинутий і прапор ***RXCn*** набуде нульового значення. Якщо встановлений біт дозволу переривання по завершенні прийому ***RXCIEn*** в регістрі **UCSRnB**, то при встановленні прапора ***RXCn*** програма переходить на вектор обробки даного переривання. Буферні регістри даних передавача і приймача USART розташовані однієї в тій же адресі в області SFR, позначеною як регістр даних **UDRn**. Якщо виконувати запис за адресою регістра **UDRn**, то записувані дані поміщаються в буферний регістр даних передавача. Аналогічно при читанні регістра **UDRn** витягується вміст буферного регістра даних приймача.

## Завдання до роботи:

1. Написати програму на мові Сі згідно з варіантом завдання.
2. Створити схему (Рис. 1) в програмі для моделювання (SimulIDE або Proteus 8). Провести моделювання написаної програми.
3. При наявності деталей зібрати схему на макетній платі та запрограмувати мікроконтролер.

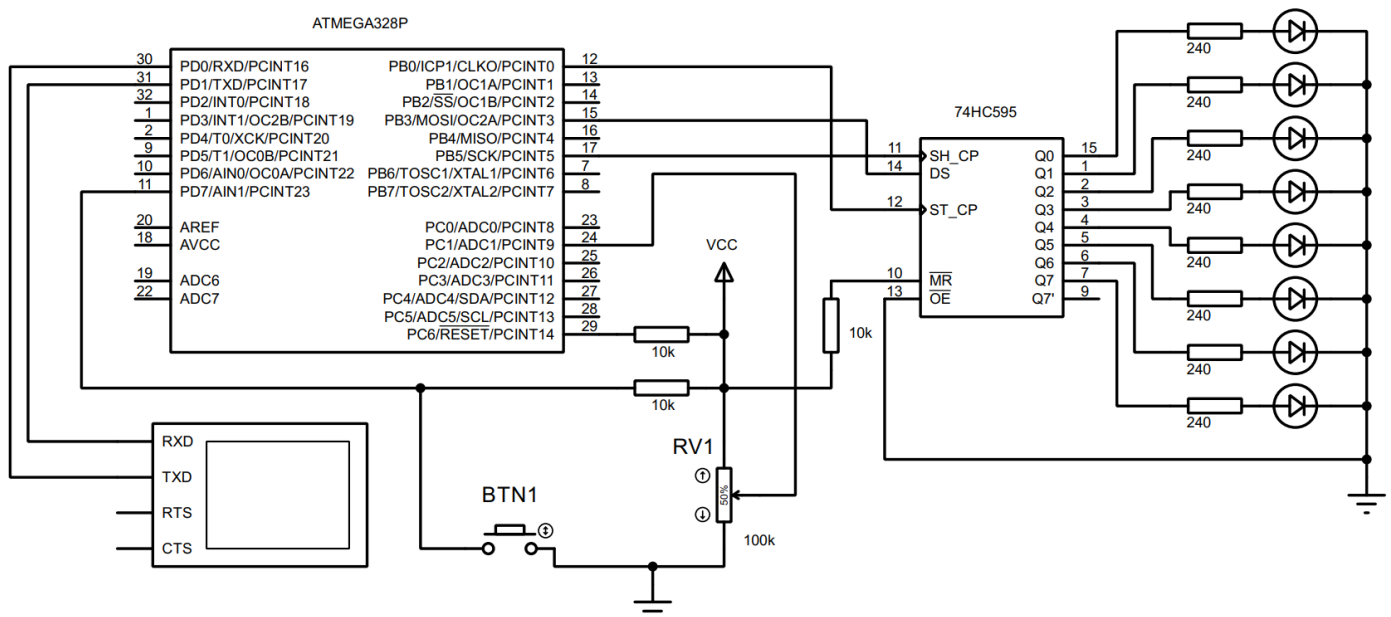


Рис 1.

Тестова програма: (Програма виводить в термінал слово **Hello!**, та реалізує ефект “echo”)

```
#include <avr/io.h>

void USART_Init(uint32_t baud)
{
    //Розрахунок швидкості (U2X = 0)
    UBRR0H = (uint8_t) ((F_CPU/(16*baud)-1)>>8);
    UBRR0L = (uint8_t) (F_CPU/(16*baud)-1);
    UCSR0A = 0;
    //Включити передавач та приймач
    UCSR0B = (1 << RXEN0) | (1<<TXEN0);
    //Налаштування фрейму передачі/прийому даних
    //8-біт даних, 1-стоп біт, без перевірки парності
    UCSR0C = (1<<UCSZ01) | (1<<UCSZ00);
}

void USART_PutChar(uint8_t data)
{
    // Чекаємо завершення передачі даних
    while (!(UCSR0A & (1<<UDRE0))){ ; }
    // Почати передачу даних
    UDR0 = data;
}

uint8_t USART_Receive(void)
{
    // Чекаємо завершення прийому даних
    while (!(UCSR0A & (1<<RXC0))){ ; }
    // Повертаємо прийняті дані
    return UDR0;
}

void print_string(const char *str)
{
    while (*str) {
        USART_PutChar(*str++);
    }
}

//=====
int main(void)
{
    USART_Init(9600);
    print_string("Hello!\r\n");

    for (;;)
    {
        USART_PutChar(USART_Receive());
    }
    return 0;
}
```

### Варіанти завдань

1. Розробити програму для схеми на Рис 1., яка при введенні цифр від 1 до 8 засвічує відповідні світлодіоди підключені до регістра зсуву (Q0..Q7), введення цифри 0 гасить світлодіоди. Якщо ввести будь-який інший символ, то в термінал виводиться справка (help). Налаштування UART: 8 –біт даних, 1 стоп-біт, швидкість 115200 бод.
2. Використовуючи схему Рис 1. Реалізувати вольтметр з виводом результату у термінал кожні 2 секунди. Вимірювальна напруга подається на вивід ADC1 із змінного резистора RV1. Налаштування UART: 8 –біт даних, 1 стоп-біт, швидкість 9600 бод.
3. Написати програму для схеми Рис 1., яка реалізує вивід відладочної інформації в термінал про стан кнопки BTN1 (коротке, довге та подвійне натиснення на кнопку). Для реалізації програми використовувати стандартні потоки вводу-виводу та швидкість UART 57600 бод.
4. (\*) Використовуючи схему Рис 1. написати програму «автомат світлових ефектів». Керування ефектами (вибір ефекту, зміна швидкості, вивід справки, ...) відбувається введенням відповідних команд в терміналі.