

УДК 004.8/681.5

DOI: [https://doi.org/10.32515/2664-262X.2022.6\(37\).1.88-99](https://doi.org/10.32515/2664-262X.2022.6(37).1.88-99)

Р.М. Минайленко, доц., канд. техн. наук, **К.О. Буравченко**, канд. техн. наук,
В.А. Резніченко, викл.

Центральноукраїнський національний технічний університет, Кропивницький, Україна
e-mail: aron70@ukr.net

Особливості програмування адаптера послідовного інтерфейсу з використанням електронного емулятора

В статті розглянуто особливості програмування адаптера послідовного інтерфейсу з використанням електронного емулятора. Показано, що програми-емулятори дають можливість провести дослідження та проаналізувати відповідні системи або пристрої.

Демонстраційний експеримент не вичерпує всіх можливостей активного сприйняття студентами досліджуваного явища і не завжди забезпечує отримання ними якісних знань, оскільки його тільки спостерігають, а не проводять самі. А тому демонстрації із залученням програм-емуляторів потрібно доповнювати виконанням студентами лабораторних робіт з їх допомогою. Програмний емулятор дозволяє проводити відповідну роботу і самостійно (позааудиторно), без залучення викладача. Це дозволяє розширити область зв'язку теорії з практикою, привчити студентів до самостійної дослідницької роботи. Крім того, можливість проводити досліди віддалено від ВНЗ вказує на перспективу використання емуляторів для дистанційного навчання.

Програмні моделі загалом дають можливість організувати якісний навчальний процес підготовки фахівців з обчислювальної техніки та комп'ютерних систем
електронний емулятор, інтерфейс, програмна модель, адаптер

Постановка проблеми. Впровадження інформаційних освітніх технологій у навчальних закладах України є одним з головних чинників у підготовці високоякісного фахівця. Найбільш характерною ознакою освіти на сучасному етапі розвитку є її інформатизація, обумовлена насамперед розповсюдженням у навчальних закладах сучасної комп'ютерної техніки та програмного забезпечення, використанням можливостей Інтернет, набуттям і накопиченням фахівцями досвіду використання інформаційних технологій (ІТ) у своїй діяльності.

Дослідження показують, що на сьогоднішній день у вітчизняній системі освіти використовуються апаратні емулятори (стенди) та програмні імітаційні моделі, призначені для вивчення обчислювальної техніки, її складових і процесів, які в ній протікають. Програмні емулятори дозволяють візуалізувати процес програмування мікросхем у спрощеному вигляді.

Програмні моделі демонструють функціональні взаємозв'язки і взаємодію вказаного пристрою, відображають результати його програмування, а також дозволяють користувачу спостерігати за всіма функціями та внутрішніми процесами.

Аналіз останніх досліджень і публікацій. Функціональне призначення універсального синхронно-асинхронного прийомопередавача (УСАПП) визначається способами його програмування. Для цього центральний процесор (ЦП) передає в УСАПП ряд керуючих слів після встановлення його в початковий стан.

Керуючі слова розділяються на дві групи:

- команди режиму;
- команди керування;

Команда режиму слідує одразу за встановленням в початковий стан, тобто появою команди RESET. Команда керування повинна слідувати за командою режиму або символами SYNC.

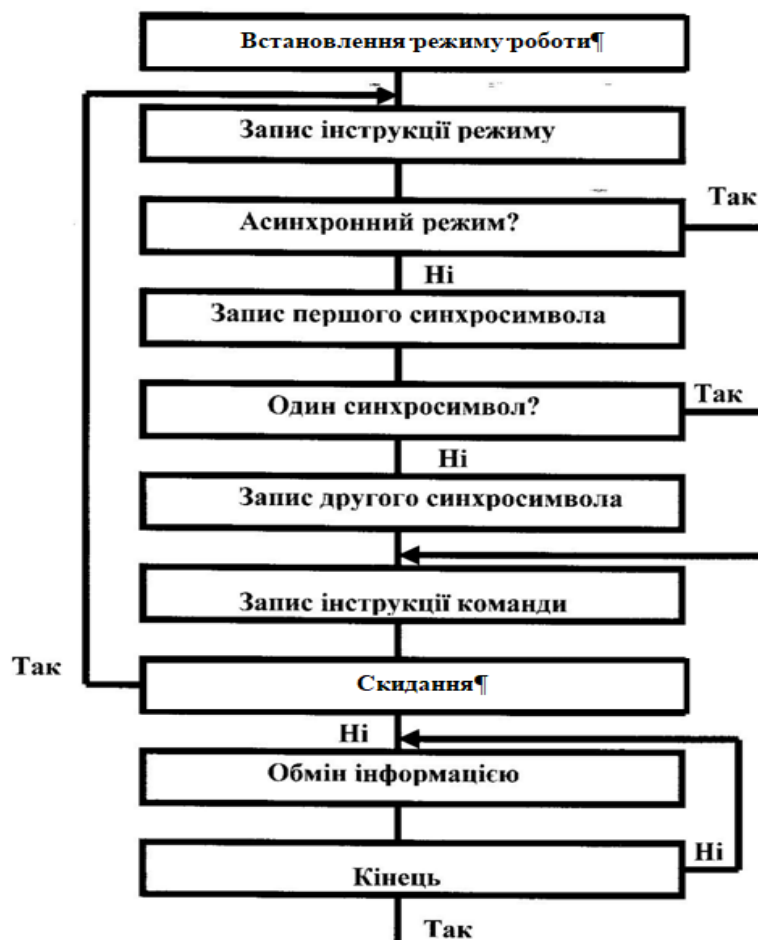


Рисунок 1 – Послідовність програмування УСАПІІ

Джерело: [2]

Команди керування можуть бути записані в будь-який час роботи.

Для повернення до команди встановлення нового режиму використовують регістр команди керування. Якщо IR у команді керування встановити в 1, то відбудеться повернення до команди режиму.

Таблиця 1 – Операції, обумовлені сигналами керування від МП

Операція	Сигнали керування			
	C/D	RD	WR	CS
Читання даних з УСАПІІ на D(7-0)	0	0	1	0
Запис даних з D(7-0) в УСАПІІ	0	1	0	0
Читання слова стану з УСАПІІ на D(7~0)	1	0	1	0
Запис керуючого слова D(7-0) в УСАПІІ	1	1	0	0
Відключення УСАПІІ від D(7-0)	X	1	1	0
Відключення УСАПІІ від D(7-0)	X	X	X	1
Примітка X — будь-який стан сигналу.				

Джерело: [2]

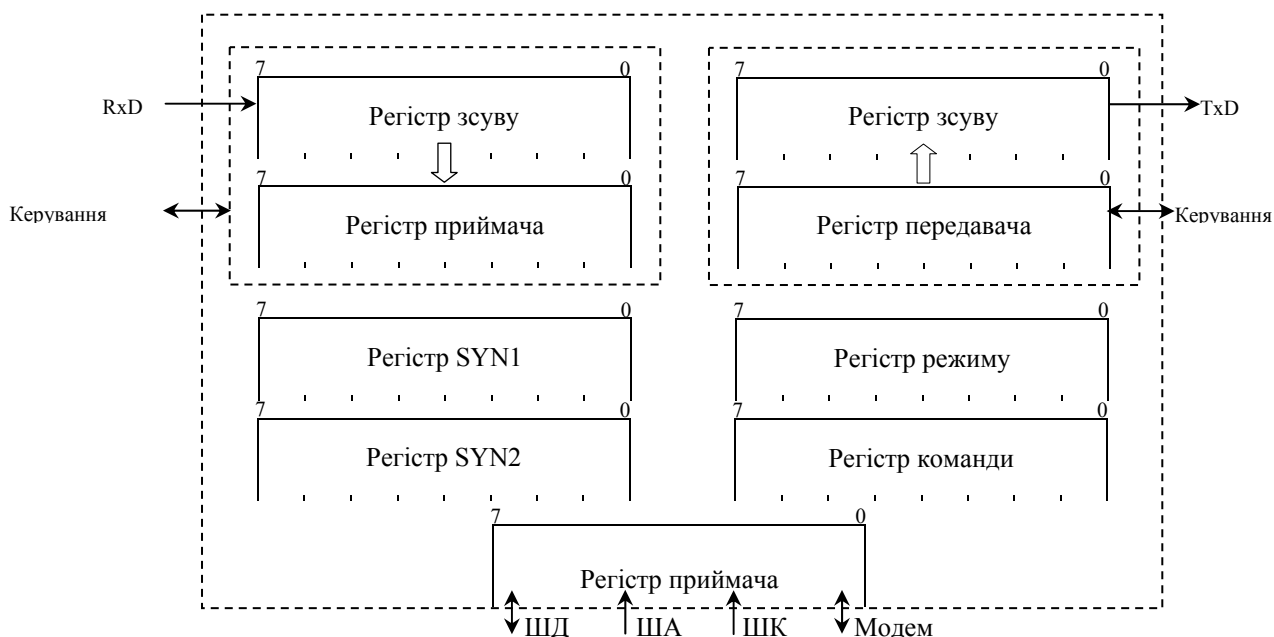


Рисунок 2 – Послідовність програмування УСАПП

Джерело: [6]

Команда режиму визначає основні робочі характеристики адаптера.

Команда керування виконує наступні функції:

- а) встановлення операції вводу або виводу;
- б) скидання тригерів прапорців помилок;
- в) керування модемом.

Читання стану.

Ця операція дозволяє ЦП в довільний час роботи зчитувати стан адаптера з метою виявлення помилок, а також сканувати запити зовнішніх пристроїв. Читання стану виконується якщо сигнал CD встановлений в стан логічної 1.

Передача/прийом даних.

Після того як слово режиму запрограмує потрібний режим адаптера і при необхідності будуть завантажені один або два синхроімпульси – адаптер готовий до обміну даними. Завантаження відповідного керуючого слова визначає передачу або приймання інформації адаптером. Рівень лог. 1 на TxRDY сигналізує ЦП про те, що адаптер готовий до прийому сигналу. Після запису символу в адаптер рівень TxRDY встановлюється в лог. 0. Адаптер може також приймати послідовні дані від модему або пристрою в/в і по завершенню прийому адаптер встановлює на RxRDY рівень лог. 1, що служить сигналом для ЦП про готовність адаптера передати йому цей символ. Адаптер не може почати передачу до тих пір поки розряд TXEN у команді керування не встановлений в лог. 1 і не отримав сигнал гашення даних.

Асинхронний режим (передача).

При передачі даних адаптер до перетвореного послідовного коду слова даних додає спочатку стартовий біт, а в кінці стоповий. Крім того якщо контроль парності передбачений у команді режиму, перед стоп-бітом встановлюється біт перевірки на парність, або непарність. Передача даних здійснюється через вихід TXD. Послідовні дані видаються передавачем на вихід по спаду TXC/ з частотою 1.16 або 1.64 від частоти TXC/. Вихід TXD після передачі слова даних і при відсутності слідуючого символа переходить в стан лог. 1, поки нові дані не надійдуть вів ЦП. У команді

керування передбачена можливість переводу виходу TXD, при відсутності даних, у стан логічного 0.

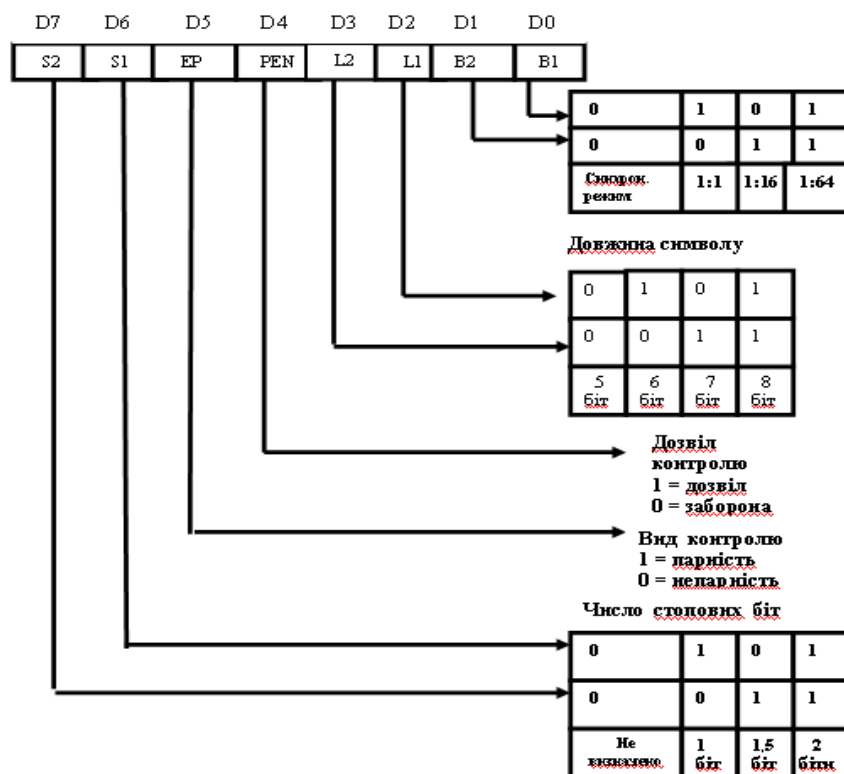


Рисунок 3 – Формат команди асинхронного режиму

Джерело: [6]

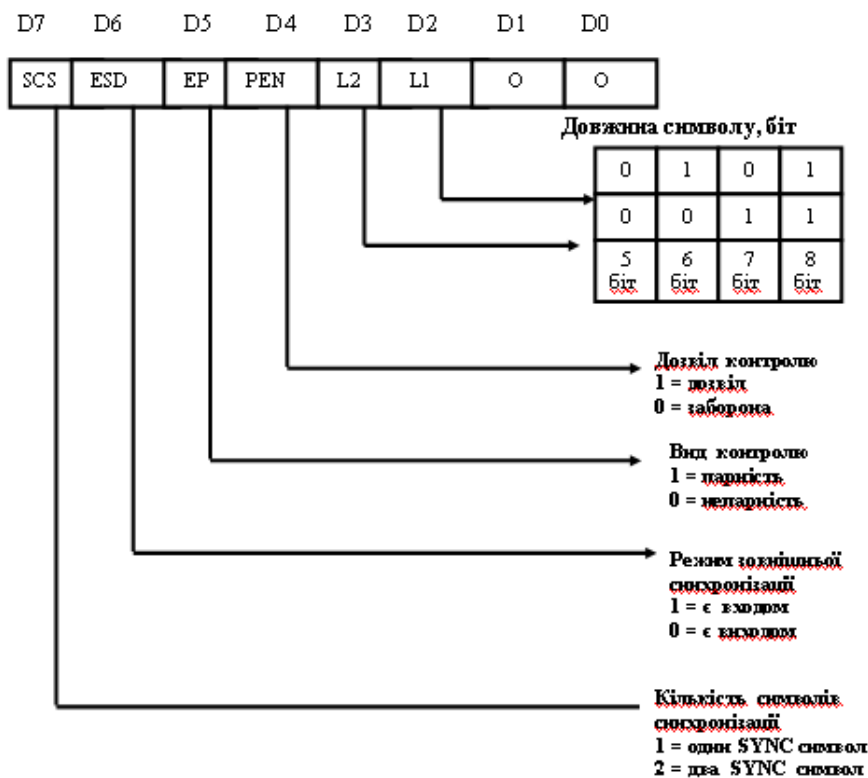


Рисунок 4-Формат команди синхронного режиму

Джерело: [6]

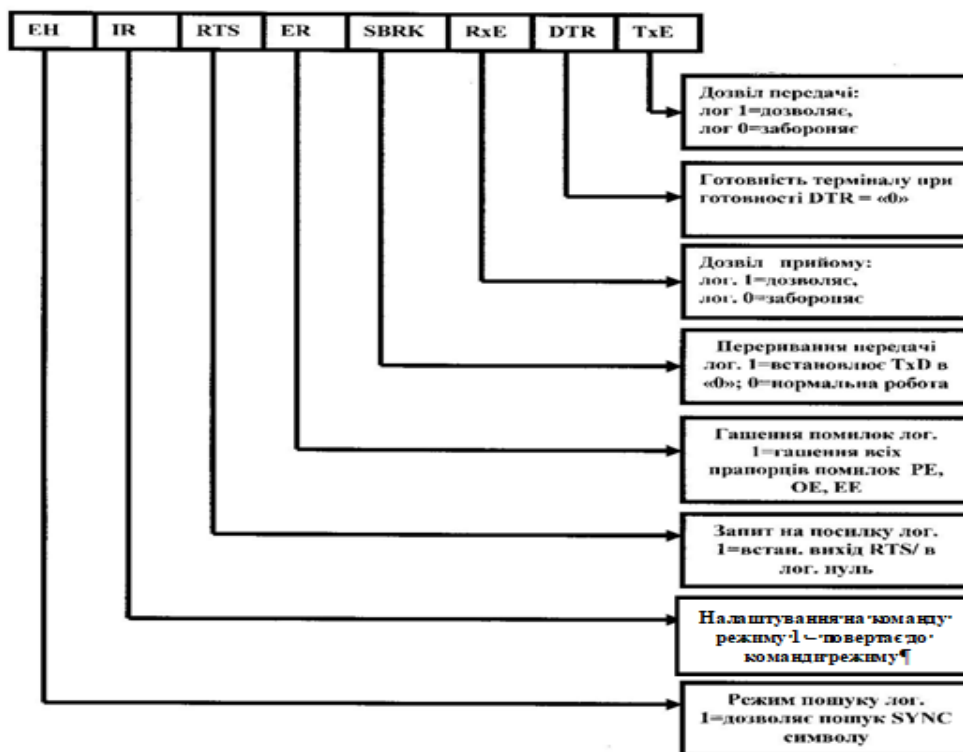


Рисунок 5 – Формат команди керування

Джерело: [6]

Асинхронний режим (прийм.)

Рівень на вході RXD знаходиться в стані логічної 1. Спад сигналу на цьому вході свідчить про появу старт-біта. Достовірність біта контролюється повторним стробуванням. Повторне знізходження рівня логічного 0 свідчить про достовірність стартowego біта. При цьому при спаданні RXD/ запускається лічильник, який відраховує задану програмно довжину слова даних, біт парності і стоп-біти. Дані приймаються в послідовному коді по фронту RXC/. При знаходженні помилки по парності встановлюється прапорець помилки по парності. Після прийняття стоп-біта адаптер здійснює передачу прийнятого символу у паралельному коді у буфер даних для передачі у ЦП. При цьому сигнал RxRDY встановлюється в логічну 1 і поточна інформація записується у буфер, стираючи попередню. Всі прапорці помилок скидаються командою керування.

Адресація портів KP580BB51

KP580BB51 ініціюється після надходження сигналу рівня логічного 0 на контакт входів мікросхеми коли адреси порту введення/виведення (в/в) на адресних лініях знаходяться в діапазоні EC..EF. Для адресації використовуються розряди з 2-го по 7-й.

Молодший розряд адреси A0 керує входом CD.

адреса пристрою в/в	команда	функція	направлення
ED або EF	виведення	команда	ЦП > УСАПП
EC або EE	виведення	дані	ЦП > УСАПП
ED або EF	введення	стан	УСАПП > ЦП
EC або EE	введення	дані	УСАПП > ЦП

Джерело: розроблено автором

Організація переривання за допомогою KP580BB51.

Для організації переривання можна використати вихід готовності приймача RxRDY. Крім цього можна використати вихід готовності передавача TxRDY, або вихід TXE.

TxRDY – логічна 1 коли адаптер готовий до приймання символів від ЦП.

RxRDY - логічна 1 коли адаптер має символ готовий до передачі в ЦП.

TXE - логічна 1 коли в буфері відсутні дані.

Виходи TxRDY, TXE керуються бітом дозволу прийому у команді керування.

Постановка завдання. Створення програм-емуляторів і послідовне їх використання в навчальному процесі дає можливість проводити дослідження та аналізувати відповідні системи або пристрої.

Демонстраційний експеримент не вичерпує всіх можливостей активного сприйняття студентами досліджуваного явища і не завжди забезпечує отримання ними якісних знань, оскільки за ним тільки спостерігають, а не проводять самі учні. Тому демонстрації із залученням програм-емуляторів потрібно доповнювати виконанням студентами лабораторних робіт з їх використанням. Програмний емулятор дозволяє проводити дослідження і самостійно (позааудиторно), тобто без залучення викладача. Це дозволяє розширити область зв'язку теорії з практикою, привчити студентів до самостійної дослідницької роботи. Крім того, можливість проводити досліди віддалено від ВНЗ вказує на перспективу використання емуляторів для дистанційного навчання.

Використання програм-емуляторів дасть можливість організувати якісний навчальний процес підготовки фахівців з обчислювальної техніки та комп'ютерних систем.

Виклад основного матеріалу. Програмований послідовний інтерфейс забезпечує передачу і приймання інформації рівнями транзисторно-транзисторної логіки (ТТЛ), або тактовими сигналами. Вибір каналів прийому/передачі здійснюється завдяки програмно керованим рівням DTR і RTS адаптера.

За допомогою емулятора програмованого адаптеру послідовного інтерфейсу I8251 прослідкуємо процес програмування пристрою. Загальний вигляд емулятора представлений на рис.6 (vv51.exe).

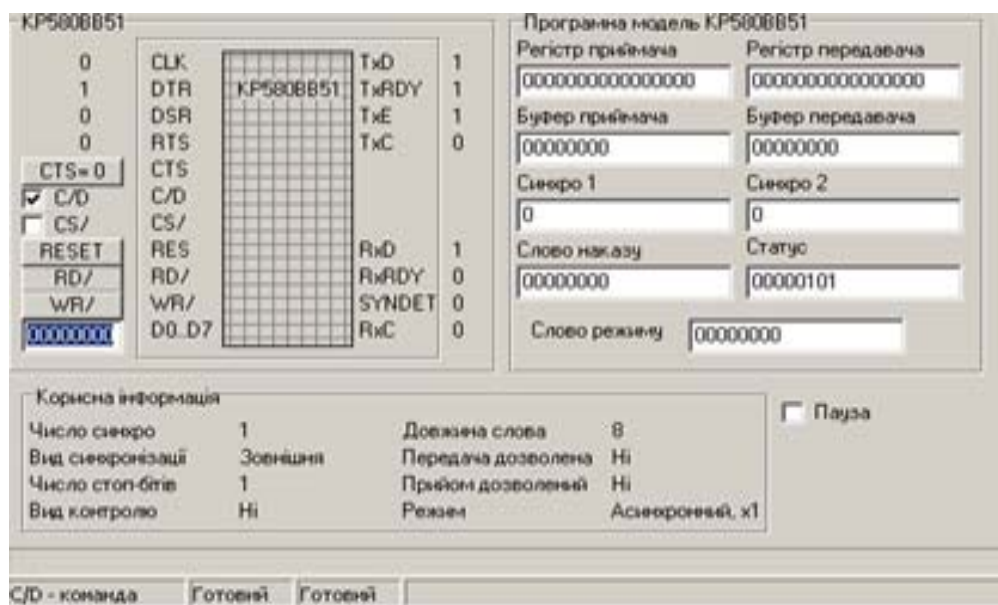


Рисунок 6 - Загальний вигляд емулятора УСАПІ I8251

Джерело: розроблено автором

адреса пристрою в/в	команда	функція	напрявлення
ED або EF	виведення	команда	ЦП > УСАПП
ЕС або EE	виведення	дані	ЦП > УСАПП
ED або EF	введення	стан	УСАПП > ЦП
ЕС або EE	введення	дані	УСАПП > ЦП

Завантаження програми до емулятора

1. Для початку роботи емулятора необхідно завантажити керуюче слово (інструкцію) режиму та керуюче слово (інструкцію) команди. В вікні емулятора натиснемо кнопку «Параметри».

2. У відповідному віконці встановимо $C\backslash D=1(\checkmark)$ та натиснемо кнопку „Reset” тим самим виконаємо скидання адаптера. Після виконаних дій з’явиться вікно , в яке повинні ввести керуюче слово вибраного режиму (рис.7):

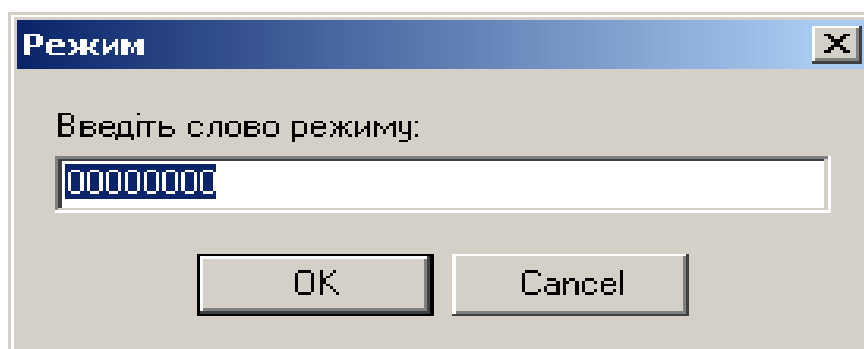


Рисунок 7 – Вікно керуюче слово вибраного режиму

Джерело: розроблено автором

3. Ввести УСР(керуюче слово режиму -) і натиснути кнопку "Ок". В загальному вікні емулятора у віконці «Слово режиму» з’явиться набрана команда.

4. У виділене віконце «D0-D7» введемо УСК (керуюче слово команди) і натиснемо кнопку "WR". Набрана команда з’явиться у віконці «Слово наказу». В вікні «Корисна інформація» відобразиться вибраний режим та параметри посилки.

5. Встановимо $C\backslash D=0$. Введемо слово даних (до віконця «D0-D7») і натиснемо кнопку "WR". Формат даних відобразиться у віконці «Буфер передавача».

5. Натиснемо кнопку CTS (встановимо $CTS=0$) таким чином дозволивши передачу. Дані послідовно будуть передаватися до зовнішнього пристрою (віконце «Регістр передавача». У вікні «Цифровий аналізатор» (рис.8) можна прослідкувати процес передачі. В вікні з зображенням мікросхеми та головними сигналами можна спостерігати за зміною сигналів в процесі роботи адаптера.

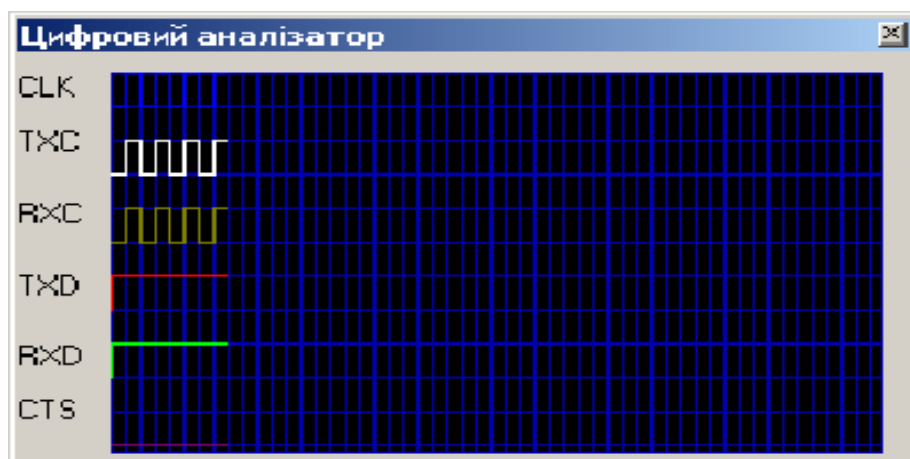


Рисунок 8 – Вікно «Цифровий аналізатор»

Джерело: розроблено автором

Використавши комплекс для програмування адаптеру загальний вигляд якого представлений на рисунку 9 можна набрати та відлагодити програму роботи пристрою. При запуску програми є можливість в покроковому режимі відслідковувати завантаження всіх команд та роботу пристрою, процес програмування якого відображується на індикаторах формату команд, даних та вхідних і вихідних сигналів мікросхеми.



Рисунок 9- Загальний вигляд комплексу програмування адаптеру

Джерело: розроблено автором

Загальний вигляд вікна компілятора показано на рисунку 10:

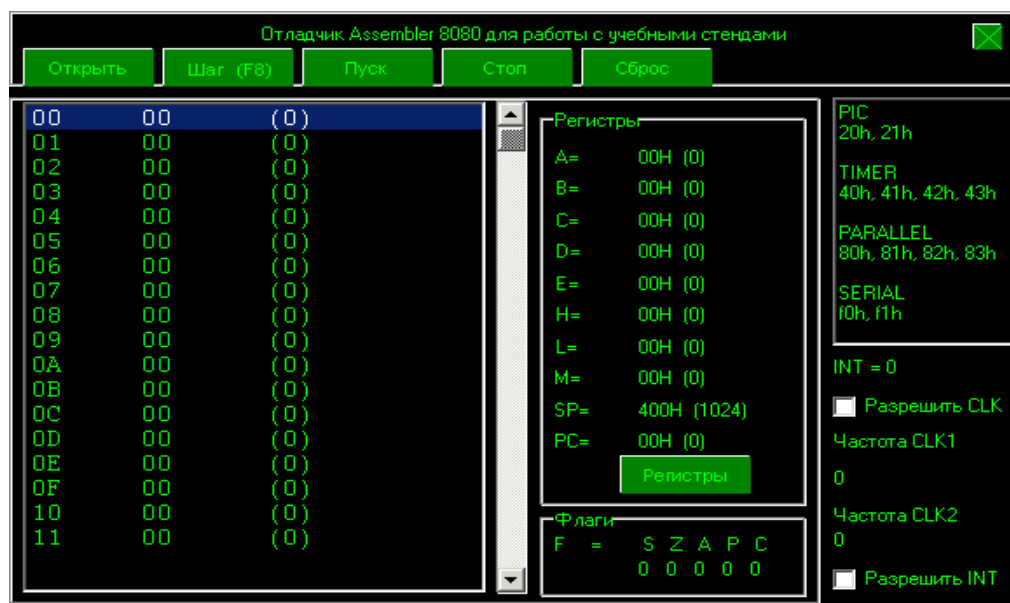


Рисунок 10 – Загальний вигляд вікна компілятора

Джерело: розроблено автором

Висновки. Впровадження інформаційних освітніх технологій у навчальних закладах України є одним з головних чинників у підготовці високоякісного фахівця. Найбільш характерною ознакою освіти на сучасному етапі розвитку є її інформатизація, обумовлена насамперед розповсюдженням у навчальних закладах сучасної комп'ютерної техніки та програмного забезпечення, використанням можливостей Інтернет, набуттям і накопиченням фахівцями досвіду використання інформаційних технологій (ІТ) у своїй діяльності.

Демонстраційний експеримент не вичерпує всіх можливостей активного сприйняття студентами досліджуваних явищ, не завжди забезпечує отримання ними дійових знань, оскільки його тільки спостерігають, а не проводять самі. А тому демонстрації із залученням програм-емуляторів потрібно доповнювати виконанням студентами лабораторних робіт з їх допомогою.

Програмний емулятор дозволяє проводити відповідну роботу і самостійно (позааудиторно), без залучення викладача. Це дозволяє розширити область зв'язку теорії з практикою, привчити студентів до самостійної дослідницької роботи. Крім того, можливість проводити досліди віддалено від ВНЗ вказує на перспективу використання емуляторів для дистанційного навчання. Програмні моделі загалом дають можливість організувати якісний навчальний процес підготовки фахівців з обчислювальної техніки та комп'ютерних систем

Список літератури

1. Бабич М.П., Жуков І.А. Комп'ютерна схемотехніка, МК-Пресс, 2004. 412с.
2. Матвієнко М. П., Розен В.П. Комп'ютерна схемотехніка: навч. посіб. К.: Видавництво "Ліра-К", 2016. 192с.
3. Бучинський М.Я., Горик О.В., Чернявський А.М., Яхін С.В. ОСНОВИ ТВОРЕННЯ МАШИН / за редакцією О.В. Горика. Харків : Вид-во «НТМТ», 2017. 448 с.
4. Рябенський В.М. Жуйков В.Я. Ямненко Ю.С. Заграничний А.В. Схемотехніка: Пристрої цифрової електроніки Електронний підручник для вищих навчальних закладів. Київ, 2016. 400 с.
5. Матвієнко М.П. Комп'ютерна схемотехніка: навч. посіб. К.: Видавництво "Ліра-К", 2013. 192 с.
6. Колонтаєвський Ю. П., Сосков А. Г. Промислова електроніка: Теорія і практикум: підручник / за ред. А. Г. Соскова. К.: Каравела, 2013. 496 с.

7. Матвієнко М.П. Основи електротехніки та електроніки: підручник. К.: Видавництво "Ліра-К" , 2016. 228 с.
8. Комп'ютерна схемотехніка та архітектура комп'ютерів. Лабораторний практикум / Єфимець В.М. та ін. К.: НАУ, 2013. 64с.
9. Комп'ютерна схемотехніка: підручник / Азаров О.Д. та ін. Вінниця: ВНТУ, 2018. 230 с
10. Комп'ютерна схемотехніка: методичні рекомендації для студентів денної та заочної форми навчання за спеціальністю 123 "Комп'ютерна інженерія"/ уклад. Сидоренко В.В., Минайленко Р.М., Михайлов С.В. Кропивницький: ЦНТУ, 2019. 72 с.
http://citforum.ck.ua/database/advanced_intro (дата звернення: 01.09.2022)
11. ХНАДУ "Файловий архів". URL: <http://files.khadi.kharkov.ua/mekhatronikitransportnikh-zasobiv.html>
Конспекти лекцій, методичні вказівки.
12. Світ електронних схем. URL : <http://ua.nauchebe.net> <http://asm.shadrinsk.net> (дата звернення: 01.09.2022)
13. Голотенко О.С. Архітектура комп'ютерних систем: конспект лекцій для студентів усіх форм навчання з курсу «Архітектура комп'ютерних систем». Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2016 . 120 с.
14. Тарарака В.Д. Т19 Архітектура комп'ютерних систем: навчальний посібник. Житомир : ЖДТУ, 2018. 383 с
15. Терещенко Т.О., Ямненко Ю.С. Сучасні напрямки комп'ютерної та мікропроцесорної техніки. Розділ 1. Основні тенденції розвитку комп'ютерної і мікропроцесорної техніки. Розділ 2. Характеристики ARM і Cortex процесорів: конспект лекцій : для студ. спеціальності 171 Електроніка, спеціалізації «Електронні компоненти та системи» . КПІ ім. Ігоря Сікорського. Київ: КПІ ім. Ігоря Сікорського, 2020. 68 с.
16. Кавун С.В., Сорбат І.В. Архітектура комп'ютерів. Особливості використання комп'ютерів в ІС : навч. посіб. Харків : Вид. ХНЕУ, 2010. 256 с.
17. Мартін Р. Чиста архітектура. Фабула, 2019. 368 с
18. Тарарака В.Д. Архітектура комп'ютерних систем: навчальний посібник. Житомир: ЖДТУ, 2018. 383 с.
19. Матвієнко М.П., Розен В.П., Закладний О.М. Архітектура комп'ютерів: навч. посіб. К. : Ліра-К, 2019. 264 с.
20. Сидоренко В.В., Минайленко Р.М., Михайлов С.В. Архітектура комп'ютерів: методичні рекомендації для студентів денної та заочної форми навчання за спеціальністю 123 "Комп'ютерна інженерія". Кропивницький: ЦНТУ, 2019. 63 с.
21. Рикалюк Р.Є., Галамага Л.Б., Селіверстов Р.Г. Лабораторний (симуляційний) практикум з курсу «Архітектура комп'ютерних систем». Ч. 2 . Львів: Видавн. центр Львів. ун-ту, 2017. 33 с
22. Комп'ютерна схемотехніка та архітектура комп'ютерів. Лабораторний практикум / В.М. Єфимець, С.В. Красовська та ін. К.: НАУ, 2013. 64с.
23. Bartlett J. Programming from the Ground Up. URL: [http://www.freebookcentre.net/ComputerScience-BooksDownload/Programming-fromthe-Ground-Up-\(J.-Bartlett\).html](http://www.freebookcentre.net/ComputerScience-BooksDownload/Programming-fromthe-Ground-Up-(J.-Bartlett).html) (дата звернення: 06.09.2022)
24. Світ електронних схем: веб-сайт. URL: <http://ua.nauchebe.net> (дата звернення: 06.09.2022)

References

1. Babich, M.P. & Zhukov, I.A. (2004). *Komp'iuterna skhemotekhnika [Computer circuit engineering]*, MK-Press [in Ukrainian].
2. Matvienko, M.P. & Rosen, V.P. (2016). *Komp'iuterna skhemotekhnika [Computer circuitry]*. Kyiv: Lira-K Publishing House [in Ukrainian].
3. Buchynskiy, M.Ya., Horyk, O.V., Chernyavskiy, A.M. & Yakhin, S.V. (2017). OSNOVY TVORENNYA MASHYN [FUNDAMENTALS OF MACHINE CREATION]. O.V. Horyk (Eds.). Kharkiv: "NTMT" Publishing House [in Ukrainian].
4. Ryabenky, V.M., Zhuykov, V.Ya., Yamnenko, Yu.S. & Zagranichnyi, A.V. (2016). Skhemotekhnika: Prystroji tsyfrovoi elektronik [Schemotechnika: Devices of digital electronics]. *Electronic textbook for higher educational institutions*. Kyiv [in Ukrainian].
5. Matvienko, M.P. (2013). *Komp'iuterna skhemotekhnika [Computer circuitry]*. Kyiv: Vydavnytstvo "Lira-K" [in Ukrainian].
6. Kolontaievs'kyj, Yu.P. & Soskov, A.H. (2013). Promyslova elektronika: Teoriia i praktykum [Industrial electronics: Theory and practice] . A.G. Soskova (Ed.). Kyiv: Karavela [in Ukrainian].
7. Matviienko, M.P. (2016). *Osnovy elektrotekhniki ta elektroniki [Basics of electrical engineering and electronics]*. Kyiv: Vydavnytstvo "Lira-K" [in Ukrainian].

8. Yefymets', V.M. et al. (2013). *Komp'iuterna skhemotekhnika ta arkhitektura komp'iuteriv. Laboratornyj praktykum* [Computer circuitry and computer architecture. Laboratory workshop] Kyiv: NAU [in Ukrainian].
9. Azarov, O.D. et al. (2018). *Komp'iuterna skhemotekhnika:[Computer circuitry]* . Vinnytsia: VNTU [in Ukrainian].
10. Sydorenko, V.V., Mynajlenko, R.M. & Mykhajlov, S.V (2019). *Komp'iuterna skhemotekhnika: metodychni rekomendatsii dlja studentiv dennoi ta zaochnoi formy navchannia za spetsial'nistiu 123 "Komp'iuterna inzheneriia"* [Computer circuit engineering: methodical recommendations for full-time and part-time students in the specialty 123 "Computer engineering". Retrieved from http://citforum.ck.ua/database/advanced_intro [in Ukrainian].
11. KhNADU . *Fajlovyj arkhiv* [File Archive]. Retrieved from <http://files.khadi.kharkov.ua/mekhatronikitransportnikh-zasobiv.html> Konspekty leksij, metodychni vkazivky [in Ukrainian].
12. *Svit elektronnykh skhem* [The world of electronic circuits]. Retrieved from <http://ua.nauchebe.net> <http://asm.shadrinsk.net> [in Ukrainian].
13. Holotenko, O.S. (2016). *Arkhitektura komp'iuternykh system: konspekt leksij dlja studentiv usikh form navchannia z kursu «Arkhitektura komp'iuternykh system»* [Architecture of computer systems: summary of lectures for students of all forms of education from the course "Architecture of computer systems"]. Ternopil' : Vyd-vo TNTU imeni Ivana Puliuia [in Ukrainian].
14. Tararaka, V.D. (2018). *T19 Arkhitektura komp'iuternykh system* [T19 Architecture of computer systems]. Zhytomyr : ZhDTU, [in Ukrainian].
15. Tereshchenko, T.O. & Yamnenko, Yu.S. (2020). *Suchasni napriamky komp'iuternoi ta mikroprotsesornoj tekhniki Rozdil 1. Osnovni tendentsii rozvytku komp'iuternoi i mikroprotsesornoj tekhniki. Rozdil 2 Kharakterystyky ARM i Cortex protsesoriv: konspekt leksij : dlja stud. spetsial'nosti 171 Elektronika, spetsializatsii «Elektronni komponenty ta systemy»* [Modern trends in computer and microprocessor technology Section 1. Main trends in the development of computer and microprocessor technology. Chapter 2 Characteristics of ARM and Cortex processors: lecture notes. [Electronic resource]: for students. specialties 171 Electronics, specializations "Electronic components and systems" /T. O. Tereshchenko, Yu.S. Yamnenko; KPI named after Igor Sikorskyi; structure]. KPI im. Ihoria Sikors'koho. Kyiv: KPI im. Ihoria Sikors'koho [in Ukrainian].
16. Kavun, S.V. & Sorbat, I.V. (2010). *Arkhitektura komp'iuteriv. Osoblyvosti vykorystannia komp'iuteriv v IS* [Computer architecture. Peculiarities of the use of computers in IS]. Kharkiv : Vyd. KhNEU, [in Ukrainian].
17. Martin, R. (2019). *Chysta arkhitektura* [Pure architecture]. Fabula, [in Ukrainian].
18. Tararaka, V.D. (2018). *Arkhitektura komp'iuternykh system* [Architecture of computer systems]: navchal'nyj posibnyk. Zhytomyr: ZhDTU [in Ukrainian].
19. Matviienko, M.P., Rozen, V.P. & Zakladnyj, O.M. (2019). *Arkhitektura komp'iuteriv* [Computer architecture]. K. : Lira-K [in Ukrainian].
20. Sydorenko, V.V., Mynajlenko, R.M. & Mykhajlov, S.V. (2019). *Arkhitektura komp'iuteriv* [Computer architecture]. Kropyvnyts'kyj: TsNTU [in Ukrainian].
21. Rykaliuk, R.Ye., Halamaha, L.B. & Seliverstov, R.H. (2017). *Laboratornyj (symuliatychnyj) praktykum z kursu «Arkhitektura komp'iuternykh system»* [Laboratory (simulation) workshop on the course "Architecture of computer systems".]. Part 2. L'viv: Vydavn. tsentr L'viv. un-tu, [in Ukrainian].
22. Yefymets', V.M., Krasovska, Ye.V. et al. (2013). *Komp'iuterna skhemotekhnika ta arkhitektura komp'iuteriv. Laboratornyj praktykum* [Computer circuitry and computer architecture. Laboratory workshop]. Kyiv: NAU [in Ukrainian].
23. Bartlett J. *Programming from the Ground Up*. Retrieved from [http:// www.freebookcentre.net/ComputerScience-BooksDownload/Programming-fromthe-Ground-Up-\(J.-Bartlett\).html](http://www.freebookcentre.net/ComputerScience-BooksDownload/Programming-fromthe-Ground-Up-(J.-Bartlett).html) [in English].
24. *Svit elektronnykh skhem: web sait* [The world of electronic circuits: web site]. Retrieved from <http://ua.nauchebe.net> [in Ukrainian].

Roman Minailenko, Assoc. Prof., PhD tech. sci., **Kostiantyn Buravchenko**, PhD tech. sci., **Vitalii Reznichenko**, Lecturer

Central Ukrainian National Technical University, Kropyvnytskyi, Ukraine

Features of Serial Interface Adapter Programming Using an Electronic Emulator

The article discusses the features of serial interface adapter programming using an electronic emulator. Emulators have been shown to enable research and analysis of relevant systems or devices.

A demonstration experiment does not exhaust all the possibilities of active perception of the studied phenomenon by students and does not always ensure that they obtain quality knowledge, since it is only

observed, and not carried out themselves. Therefore, demonstrations involving emulator programs should be supplemented by students performing laboratory work with their help. The software emulator allows you to carry out relevant work independently (outside the classroom), without the involvement of a teacher. This makes it possible to expand the field of connection between theory and practice, to accustom students to independent research work. In addition, the possibility of conducting experiments remotely from educational institutions points to the prospect of using emulators for distance learning.

The implementation of information and educational technologies in educational institutions of Ukraine is one of the main factors in training a high-quality specialist. The most characteristic feature of education at the current stage of development is its informatization, due primarily to the spread of modern computer technology and software in educational institutions, the use of Internet capabilities, the acquisition and accumulation of experience by specialists in the use of information technologies (IT) in their activities.

Software models in general make it possible to organize a high-quality educational process of training specialists in computing and computer systems.

electronic emulator, interface, software model, adapter

Одержано (Received) 14.08.2022

Прорецензовано (Reviewed) 25.08.2022

Прийнято до друку (Approved) 26.09.2022