

# 1 Лабораторна робота № 1.

## Ознайомлення з середовищем Micro-Cap. Дослідження логічних елементів

### 1.1 Мета

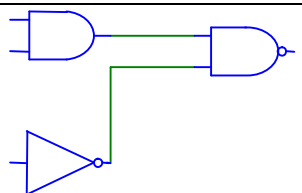
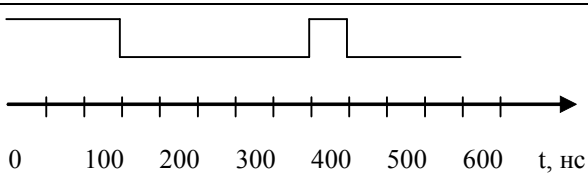
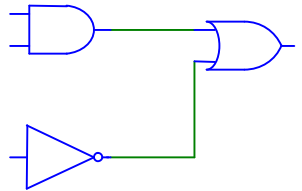
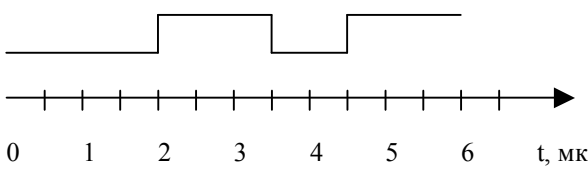
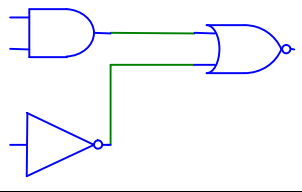
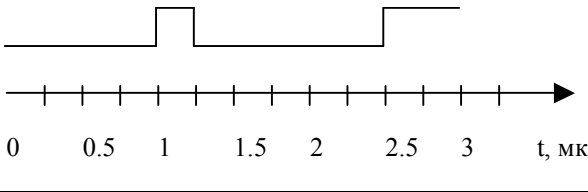
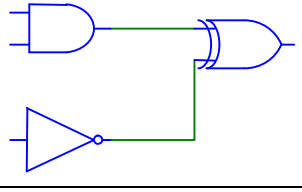
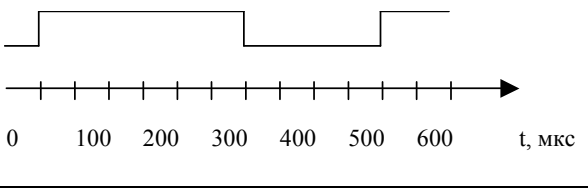
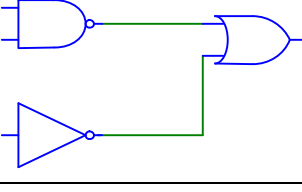
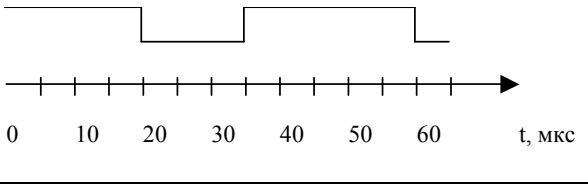
Отримати основні навички роботи з середовищем симуляції цифрових та аналогових схем Micro-Cap. Засвоїти основні типи логічних елементів.

### 1.2 Завдання

В середовищі Micro-Cap 8 скласти схему згідно індивідуального завдання, подати на входи схеми цифрові сигнали, отримати часові діаграми сигналів на входах і виході схеми.

В табл.1.1 входи схеми нумеруються зверху вниз. Сигнал на першому вході – це постійний логічний рівень (0 або 1), на другому – імпульсний сигнал заданої частоти, на третьому – задана послідовність логічних рівнів 0 та 1.

Таблиця 1.1 – Варіанти індивідуальних завдань

| Вар. | Схема   | Сигнал на вході 1 | Сигнал на вході 2 | Сигнал на вході 3  |
|------|---|-------------------|-------------------|--|
| 1.   |   | 1                 | 10 МГц            |   |
| 2.   |  | 1                 | 1МГц              |  |
| 3.   |  | 1                 | 2МГц              |  |
| 4.   |  | 1                 | 10кГц             |  |
| 5.   |  | 1                 | 100кГц            |  |

|     |  |   |         |  |
|-----|--|---|---------|--|
| 6.  |  | 1 | 200кГц  |  |
| 7.  |  | 1 | 250кГц  |  |
| 8.  |  | 1 | 400кГц  |  |
| 9.  |  | 0 | 500кГц  |  |
| 10. |  | 0 | 2,5МГц  |  |
| 11. |  | 1 | 50кГц   |  |
| 12. |  | 0 | 40кГц   |  |
| 13. |  | 0 | 25кГц   |  |
| 14. |  | 0 | 20кГц   |  |
| 15. |  | 0 | 12,5кГц |  |

### 1.3 Короткі теоретичні відомості

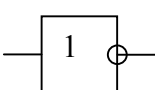

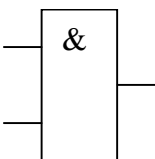

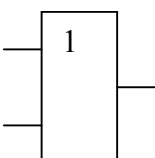

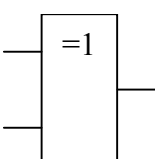

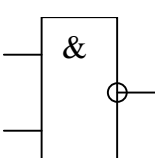

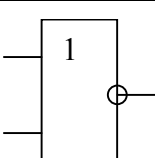

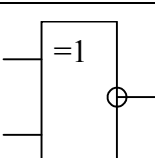

#### 1.3.1 Логічні елементи

Теоретичною основою цифрової техніки є алгебра логіки (булева алгебра), що оперує логічними змінними. Логічна змінна – це змінна, що може приймати лише два значення – 0 та 1. В цифрових схемах значенню «0» найчастіше відповідає низький рівень напруги, а значенню «1» - високий.

Логічний елемент реалізує деяку логічну функцію між значеннями цифрового сигналу на своїх входах. Логічні функції зручно описувати таблицями істинності, які показують значення на виході схеми для всіх можливих комбінацій на входах. Якщо вхід один, можливих комбінацій дві (0 та 1), для двох входів можливих комбінацій є 4 (00, 01, 10, 11).

Базовими логічними функціями є: інверсія (логічне НЕ), логічне І, логічне АБО та виключне АБО. Решта елементів утворюються як комбінація базових елементів з інверсією (І-НЕ, АБО-НЕ, виключне АБО-НЕ).

Таблиця 1.2 – Логічні елементи

| Назва логічного елемента | Позначення  |   | Логічна функція   | Таблиця істинності |   |   | Назва в розділі Standard Gates |
|--------------------------|---|---|---|--------------------|---|---|--------------------------------|
|                          | ДСТУ  | ANSI  |   | X                  | Y | Z |                                |
| НЕ (інвертор)            |    |    | $\begin{array}{c} X \\ \boxed{1} \\ Y \\ Y = \bar{X} \end{array}$                     | X                  | Y |   | Inverters                      |
|                          |   |   |   | 0                  | 1 |   |                                |
|                          |   |   |   | 1                  | 0 |   |                                |
| І                        |   |   | $\begin{array}{c} X \\ \boxed{\&} \\ Y \\ Z \\ Z = X \wedge Y \end{array}$            | X                  | Y | Z | And Gates                      |
|                          |   |   |   | 0                  | 0 | 0 |                                |
|                          |   |   |   | 0                  | 1 | 0 |                                |
|                          |   |   |   | 1                  | 0 | 0 |                                |
|                          |   |   |   | 1                  | 1 | 1 |                                |
| АБО                      |  |  | $\begin{array}{c} X \\ \boxed{1} \\ Y \\ Z \\ Z = X \vee Y \end{array}$               | X                  | Y | Z | Or Gates                       |
|                          |   |   |   | 0                  | 0 | 0 |                                |
|                          |   |   |   | 0                  | 1 | 1 |                                |
|                          |   |   |   | 1                  | 0 | 1 |                                |
|                          |   |   |   | 1                  | 1 | 1 |                                |
| ВИКЛЮЧНЕ АБО             |  |  | $\begin{array}{c} X \\ \boxed{=1} \\ Y \\ Z \\ Z = X \oplus Y \end{array}$            | X                  | Y | Z | Xor Gates                      |
|                          |   |   |   | 0                  | 0 | 0 |                                |
|                          |   |   |   | 0                  | 1 | 1 |                                |
|                          |   |   |   | 1                  | 0 | 1 |                                |
|                          |   |   |   | 1                  | 1 | 0 |                                |
| І-НЕ                     |  |  | $\begin{array}{c} X \\ \boxed{\&} \\ Y \\ Z \\ Z = \overline{X \wedge Y} \end{array}$ | X                  | Y | Z | Nand Gates                     |
|                          |   |   |   | 0                  | 0 | 1 |                                |
|                          |   |   |   | 0                  | 1 | 1 |                                |
|                          |   |   |   | 1                  | 0 | 1 |                                |
|                          |   |   |   | 1                  | 1 | 0 |                                |
| АБО-НЕ                   |  |  | $\begin{array}{c} X \\ \boxed{1} \\ Y \\ Z \\ Z = \overline{X \vee Y} \end{array}$    | X                  | Y | Z | Nor Gates                      |
|                          |   |   |   | 0                  | 0 | 1 |                                |
|                          |   |   |   | 0                  | 1 | 0 |                                |
|                          |   |   |   | 1                  | 0 | 0 |                                |
|                          |   |   |   | 1                  | 1 | 0 |                                |
| ВИКЛЮЧНЕ АБО-НЕ          |  |  | $\begin{array}{c} X \\ \boxed{=1} \\ Y \\ Z \\ Z = X \oplus Y \end{array}$            | X                  | Y | Z | Xnor Gates                     |
|                          |   |   |   | 0                  | 0 | 1 |                                |
|                          |   |   |   | 0                  | 1 | 0 |                                |
|                          |   |   |   | 1                  | 0 | 0 |                                |
|                          |   |   |   | 1                  | 1 | 1 |                                |

В вітчизняній літературі та проектній документації (це стосується також країн СНД) використовується позначення цифрових елементів згідно державного стандарту (ДСТУ, рос. ГОСТ), в зарубіжних джерелах та ряді зарубіжних програмних продуктів – як правило, згідно стандарту ANSI (скорочення від American National Standards Institute).


### 1.3.2 Робота в середовищі Micro-Cap 8



При запуску Micro-Cap автоматично створюється новий файл з розширенням .cir, в якому можна створити цифрову схему.

Для вставки цифрового елемента необхідно зайти в меню **Компоненты/Digital Primitives/**. В цьому меню можна вибрати ряд компонентів, з яких на даній лабораторній роботі використовуватимуться наступні:

- Standard gates - логічні елементи;
- Stimulus generators - генератор цифрових сигналів з заданими характеристиками.

Після вибору необхідного компонента в меню він вставляється на робоче поле натисканням лівої кнопки миші. При цьому з'являється вікно задання властивостей компонента. Наприклад, для логічних елементів слід задати часову модель: D0\_GATE (ідеальний логічний елемент – без затримки спрацювання) чи DLY\_TTL (ТТЛ-елемент), вибравши її зі списку доступних моделей.

Для вставки провідників необхідно встановити режим Wire, натиснувши кнопку . Провідник проводиться між двома довільними точками мишею при натисненій лівій кнопці.

Для виділення будь-яких елементів слід встановити режим Select, натиснувши кнопку  або пробіл. Елементи переміщуються мишею. Змінити властивості компонента можна двічі клацнувши на ньому мишкою. Слід клацати саме на компоненті, а не його назві (інакше буде виділятися і редагуватися назва, а не властивості компонента). Бажано натискати клавішу  або пробіл після закінчення вставки компонентів, щоб не розмножувати компоненти випадковим клацанням мишки на робочому полі.

Поворот елемента на 90° здійснюється натисканням правої кнопки миші, коли компонент вибраний в режимі Select при натисненій лівій кнопці.

Для видалення елемента чи провідника слід виділити його, після чого натиснути DEL на клавіатурі.

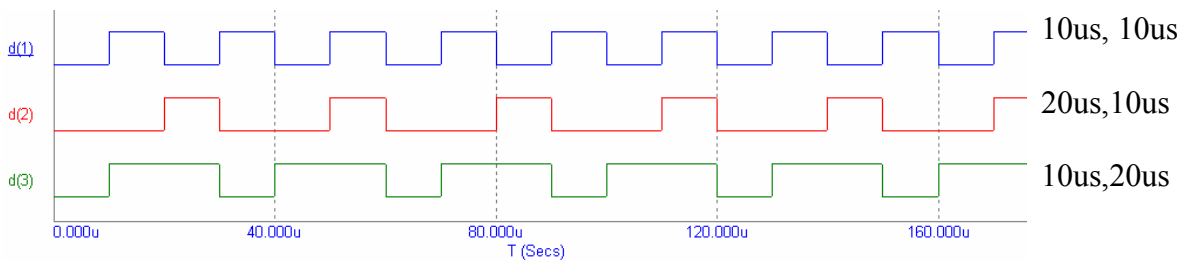
Для редагування можна також використовувати стандартні комбінації Ctrl+C (копіювання в буфер), Ctrl+V (вставка з буфера), Ctrl+X (видалення в буфер). Як правило, при копіюванні номери елементів змінюються коректно, однак бажано слідкувати, щоб у схемі не було двох елементів з однаковою назвою (це призведе до помилок при симуляції).

Для подачі на вхід схеми логічних рівнів та імпульсних послідовностей використовуються компоненти із меню **Компоненты / Digital Primitives / Stimulus Generators**.

При заданні часових параметрів для позначення часу в наносекундах використовується **ns**, мікросекундах **us**, мілісекундах **ms**, секундах **s**. Якщо не вказати одиницю виміру, час буде заданий в секундах.

Компонент DClock використовується для формування імпульсної послідовності заданої частоти. Параметри Param:Zerowidth (тривалість нульового рівня) та Param:Onewidth (тривалість одиничного рівня) задають параметр, що називається щільністю (рос. «скважность») - відношення періоду послідовності до тривалості одиничного рівня. Сума Param:Zerowidth та Param:Onewidth дорівнює періоду імпульсної послідовності.

На рис. 2.1 показано графіки імпульсних послідовностей при різних значеннях Param:Zerowidth, Param:Onewidth.



**Рисунок 1.1 – Імпульсні послідовності з різними щільностями**

Компонент *Stim1* зручно використовувати для формування довільної послідовності логічних рівнів на вході схеми. Для цього компонент необхідно запрограмувати: в полі *Command* задати довільне ім'я і у нижньому вікні після директиви *.define* ввести послідовність команд, які визначають послідовність зміни логічних рівнів на виході компонента.

Формати команд наведено в табл. 1.3. Кожна команда пишеться з нового рядка. Пробіл між числом і одиницею виміру часу не ставиться.

Знак + перед міткою часу задає час відносно початку означення команди, ++ задає час відносно попередньої команди.

Таблиця 1.3 – Формати команд програмування компонентів *Stimulus*


| Формат команди                        | Принцип роботи   | Приклади  |
|---------------------------------------|--|---|
| <time> <value>                        | встановити значення виходу в момент часу <i>time</i> у значення <i>value</i> (0 або 1)   | +0NS 1<br>++10MS 0                                  |
| LABEL=<label name>                    | задання мітки з ім'ям <i>label name</i>  | label = m1  |
| <time> GOTO <label name><br><n> TIMES | цикл з поверненням на мітку <i>label name</i> n разів. Значення n = -1 задає вічний цикл | ++5ns goto m1 8 times<br>+10us goto label2 -1 times |

Таблиця 1.4 - Приклади програмування компонента *Stim1*

| Означення компонента   | Часова діаграма | Примітки   |
|--|-----------------|--|
| .define IN1<br>+0us 1<br>+10us 0<br>+20us 1<br>+40us 0<br>+45us 1                    |                 | Час задається відносно початку означення   |
| .define IN2<br>++0us 1<br>++10us 0<br>++10us 1<br>++20us 0<br>++5us 1                |                 | Час задається відносно попередньої команди   |
| .define IN3<br>+0us 0<br>+LABEL=BEG<br>++5us 1<br>++5us 0<br>++5us goto beg -1 times |                 | Частота імпульсної послідовності<br>$f = \frac{1}{T_0 + T_1}$<br>де $T_0$ , $T_1$ – тривалості нульового і одиничного рівнів |

MicroCap автоматично присвоює всім вузлам унікальні номери. Для відображення

номерів вузлів слід натиснути кнопку .

Кожному вузлу можна присвоїти унікальне символічне ім'я. Для цього слід вибрати текстовий режим кнопкою , клацнути мишею на вузлі і ввести ім'я вузла. Використання символічних імен корисне, якщо в схему часто вносяться зміни (при цьому можуть змінюватись номери вузлів, і їх необхідно буде кожен раз міняти у параметрах часових діаграм).

Для виводу часових діаграм необхідно натиснути Alt+1 або вибрати в меню **Анализ/Переходные процессы...**. З'явиться вікно, в якому слід задати параметри часової діаграми:

1) time range <часовий інтервал>

Задається виходячи із тривалості сигналів, що аналізуються. Наприклад, якщо аналізується імпульсна послідовність з частотою 200 кГц, то, очевидно, час аналізу повинен бути не менше 1-2 періодів, тобто 5-10 мікросекунд.

2) Параметри графіка:

**P** 1 - номер графіка

**X Expression** T - величина по осі X

**Y Expression** d(<номер вузла або його символічне ім'я>)- величина по осі Y

3) Бажано відмітити перемикач **Автомасштаб** для автоматичного масштабування графіків (інакше час треба задавати в полі X Range, межі амплітуди сигналу – в полі Y Range).

Кнопкою **Добавить** додаються вузли для виводу графіків.

Кнопкою **Запуск** здійснюється вивід часових діаграм.

#### 1.4 Порядок виконання роботи

1. Скласти схему згідно завдання з використанням табл.1.1 для ідентифікації елементів. Для першого логічного елемента встановити часову модель DO\_GATE, після цього для решти елементів ця модель буде вибрана автоматично.

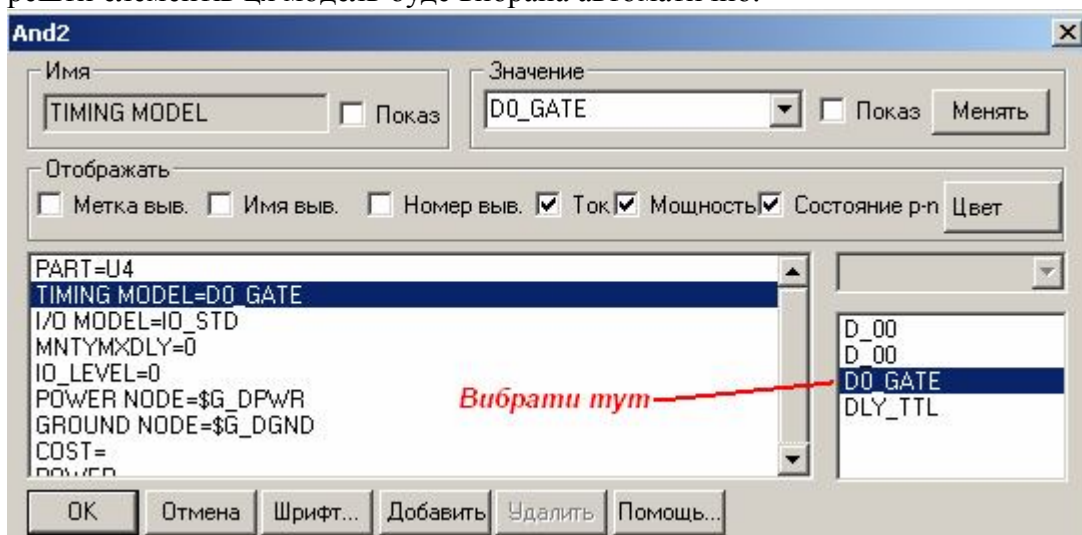
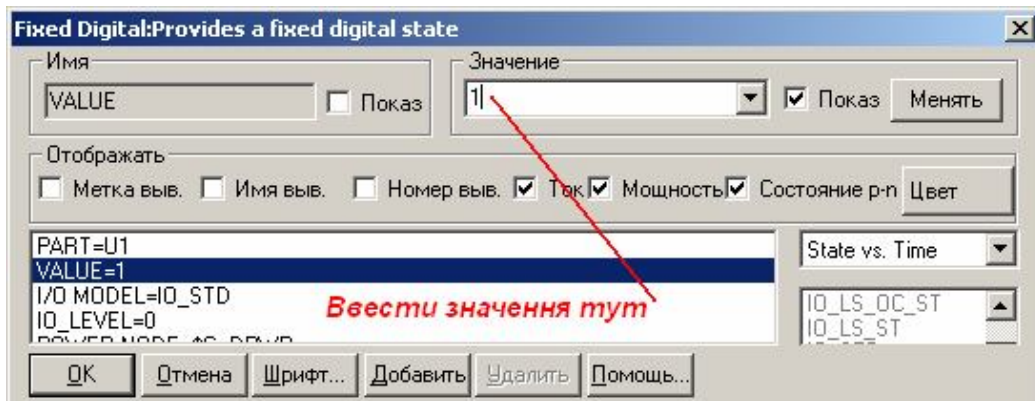


Рисунок 1.2 – Вибір часової моделі логічного елемента

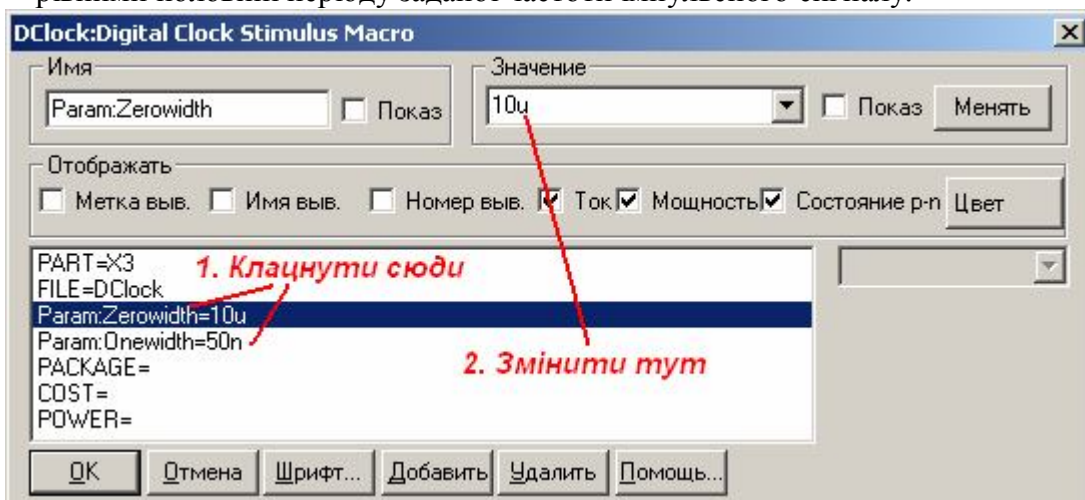
2. Використовуючи компоненти Stimulus, подати на вхід 1 фіксований логічний рівень, на вхід 2 – імпульсну послідовність з заданою частотою, на вхід 3 – задану послідовність логічних рівнів.

2.1. До першого входу приєднати компонент Fixed Digital. В полі Value задати 0 або 1 згідно завдання.



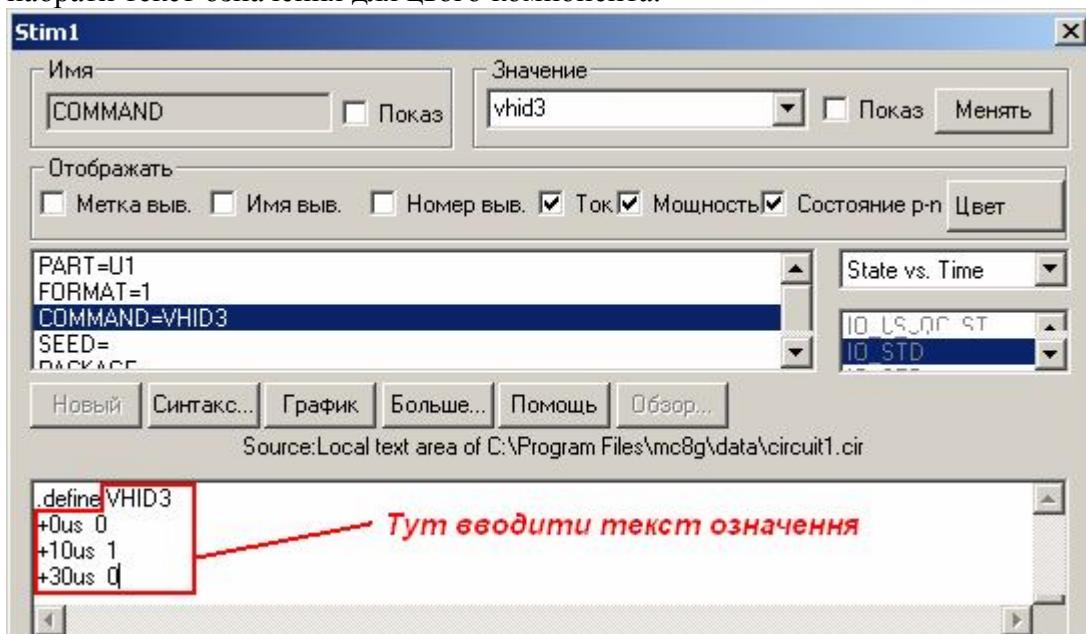
**Рисунок 1.3 – Задання фіксованого логічного рівня**

2.2. До другого входу приєднати компонент DClock. Параметри Param:Zerowidth (довжина нульового рівня), Param:Onewidth (довжина одиничного рівня) встановити рівними половині періоду заданої частоти імпульсного сигналу.



**Рисунок 1.4 – Задання параметрів компонента DClock**

2.3. До третього входу приєднати компонент Stim1. У формі властивостей компонента у віконечку внизу після слова define ввести довільне ім'я і з нового рядка набрати текст означення для цього компонента.



**Рисунок 1.5 – Програмування компонента Stim1**

Кожен рядок в означенні задає зміну рівня сигналу у відповідний момент часу, наприклад, якщо сигнал в момент часу 10мкс змінив свій рівень з 0 на 1, можна написати:

+10us 1

або, наприклад, при використанні відносного формату часу, якщо одиничний рівень сигналу тривав 5 мкс, а потім змінився на 0, можна написати:

++5us 0

3. Зберегти файл із схемою.

4. Вивести часові діаграми сигналів на входах 2 і 3 та на виході схеми.

4.1. Дати входам і виходам символічні імена. Для цього натиснути кнопку **T**, клацнути мишкою на вузлі, який слід назвати, і ввести назву у віконечку, яке відкриється.

4.2. Натиснути Alt+1 або вибрати в меню **Анализ/Переходные процессы...**. Якщо у схемі або в тексті означення компонента Stim1 є помилки (наприклад, кілька елементів мають однакову назву, або зайві пробіли в означенні), буде виведено повідомлення про помилку. Якщо помилок немає, буде виведено вікно параметрів аналізу.

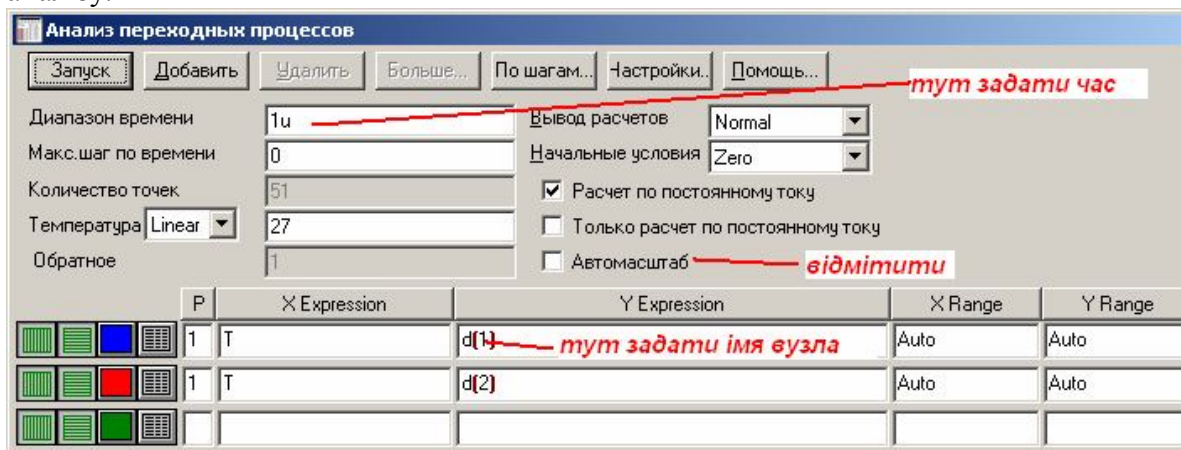


Рисунок 1.6 – Задання параметрів аналізу

4.3. Задати часовий інтервал (**Диапазон времени**), не менший за максимальний час у завданні (тривалість одного-двох періодів частоти на вході 2 або максимальний час на графіку для входу 3).

4.4. У віконці "Y Expression" задати імена вузлів для виводу в форматі d(<ім'я вузла>). Нові графіки додаються кнопкою **Добавить**.

4.5. Встановити перемикач **Автомасштаб**

4.6. Натиснути кнопку **Запуск**

5. Проаналізувати правильність отриманих графіків. На вході 2 має бути імпульсна послідовність на зразок показаної на рис. 1.2 (перший графік), на вході 3 – часова діаграма, аналогічна заданій в завданні, на виході – сигнал, що є деякою комбінацією цих двох сигналів. Якщо в одному з вузлів сигнал є постійним, можливі причини цього наступні:

- заданий надто малий час для виводу графіків;
- відсутній зв'язок між елементами схеми;
- не заданий сигнал на одному з входів схеми.

6. Для повернення до вікна із схемою закрити вікно аналізу часових діаграм (якщо його просто згорнути, схему можна буде тільки переглянути, але не можна буде редагувати).

7. Скласти цю саму схему з елементів, позначених згідно ГОСТ. Ці елементи знаходяться в меню **Компоненты / Russian Digital / И, или, не**, а також **Компоненты / Russian Digital / Буферные элементы** (там знаходяться елементи Виключне АБО).



Примітка. Графіки і схеми можна копіювати в буфер обміну (Редактирование/Копировать в буфер обмена) в форматі BMP або WMF, EMF. Після цього їх можна вставити, наприклад, в документ Word.

#### 1.5 Звіт повинен містити:

1. Завдання (з вказанням номеру варіанта)
2. Схему у позначеннях згідно стандарту ANSI
3. Схему у позначеннях згідно ДСТУ
4. Значення параметрів і означення компонентів DClock та Stim1 з поясненнями
5. Часові діаграми сигналів на входах 2 і 3 та на виході схеми
6. Висновки

#### 1.6 Контрольні питання

1. Основні прийоми роботи в середовищі Micro-Cap.
2. Відмінність між цифровими і аналоговими сигналами. Основні параметри цифрових сигналів.
3. Логічні елементи: умовні позначення, таблиці істинності логічних функцій.
4. Визначити стан виходу схеми на логічних елементах при заданих вхідних сигналах.
5. Намалювати часову діаграму в проміжній точці досліджуваної схеми або на виході довільного цифрового елемента при заданих часових діаграмах на входах.