УДК 004.942

[0000-0002-2231-2529] **М. А. Мірошник**[1], *д-р техн. наук, професор,*
e-mail: m.miroshnyk@karazin.ua
[0000-0003-1071-3445] **О. С. Шкіль**[2], *канд. техн. наук, доцент,*
e-mail: oleksandr.shkil@nure.ua
[0000-0002-6652-1840] **Д. Ю. Рахліс**[2], *канд. техн. наук, доцент,*
e-mail: dariia.rakhlis@nure.ua
[0009-0007-0799-6604] **К. Ю. Пшеничний**[2], *аспірант,*
e-mail: kyrylo.pshenychnyi@nure.ua
[0000-0001-5702-9611] **А. М. Мірошник**[2], *асистент,*
e-mail: anatolii.miroshnyk@nure.ua
[1]Харківський національний університет імені В. Н. Каразіна
майдан Свободи, 4, м. Харків, 61000, Україна
[2]Харківський національний університет радіоелектроніки
просп. Науки, 14, м. Харків, 61166, Україна

МОДЕЛЬ ОБРОБКИ ПОДІЙ ДЛЯ МОДЕЛЮВАННЯ ПРИСТРОЇВ ЛОГІЧНОГО КЕРУВАННЯ РЕАЛЬНОГО ЧАСУ

У статті проаналізовано питання моделювання пристроїв реального часу за допомогою шаблону кінцевих автоматів мовою опису апаратури. Проаналізовано класифікацію подій як моделей взаємодії кінцевого автомата з зовнішнім середовищем. Виокремлено новий вид подій та клас пристроїв, функціональність яких залежить від такого роду чинників. Показано, що такі події є поширеними у різних цифрових пристроях, приладах та системах реального часу. Запропоновано новий вид автоматного переходу для темпорального графа переходів як канонічної моделі при проектуванні цифрових систем. Розглянуті моделі проілюстровано за допомогою моделювання та аналізу часових діаграм. У роботі вирішено питання проєктування часових керуючих автоматів у системах логічного управління реального часу. Розглянуто питання систем управління з обробкою зовнішніх подій, які мають тривати визначений час.

Ключові слова: *цифрові пристрої, моделювання, моделі реального часу, сигнали, події, мовні моделі часових автоматів.*

Вступ. Логічні системи управління представляють значний вузол будь-якої цифрової системи. Такі системи використовують двійковий алфавіт для визначення поведінки блока керування. Відомо, що шаблон кінцевого автомата (Finite State Machine) є поширеною моделлю для таких систем. Варто зауважити, що кінцевий автомат є математичною абстракцією, яка може бути представлена різними способами, наприклад, таблицею переходів станів, діаграмою станів (графом переходів), граф-схемою алгоритму (ГСА) тощо. При використанні автоматного шаблону важливо розуміти, що поведінка цільової системи залежить від подій, що відбуваються в зовнішньому середовищі, та використовується для моделювання переходу з одного стану FSM до іншого. В кінцевому автоматі події використовуються для моделювання певної зовнішньої дії, яка викликає перехід з одного стану в інший. До подій відносяться сигнали, виклики, закінчення певного проміжку часу або зміна стану. Події можуть бути синхронними і асинхронними; їх вираження у кінцевій моделі – одна зі складових процесу проектування.

Моделювання пристроїв реального часу вимагає визначення поняття часу, в якому працюють модель та кінцеві пристрої. Цикли синхросигналу визначають машинний (автоматний) час, протягом якого функціонує автомат. Однак пристрої реального часу працюють у метричному часі. Інакше кажучи, стан таких пристроїв залежить як від вхідних сигналів, так і від часу, протягом якого ці сигнали обробляються. Таким чином, виникає вимога виразити метричний час у термінах автоматних тактів, оскільки переходи між станами автомата безпосередньо залежать від аспекту часу. З другого боку, необхідно виразити часові обмеження на графі переходів як початкової математичної моделі проєктування.

DOI: 10.24025/2306-4412.2.2023.274840

В [1] показано, що будь-який цифровий пристрій можна представити у вигляді двох основних компонентів – операційного та керуючих автоматів. Така форма представлення операційного пристрою (ОП) базується на принципі мікропрограмного управління, який полягає у наступному:

– будь-яку операцію пристрою можна представити як послідовність елементарних логіко-арифметичних операцій за вхідними словами;

– логічні умови визначають порядок виконання цих операцій;

– представлення алгоритму в термінах мікрооперацій та логічних умов називається мікропрограмою;

– мікропрограма є формою представлення функціональності пристрою, на основі якої визначається структура і порядок функціонування пристрою в часі.

Операційний автомат (ОА) виконує зберігання слів інформації, виконання мікрооперацій алгоритму, виз над словами та обчислення значення логічних умов, які необхідні для виконання алгоритму. Операційний пристрій визначається такими дискретними множинами: $Y = \{y_1, y_m\}$ визначає мікрооперації; $X = \{x_1, x_l\}$ визначає логічні умови, які виникають під час виконання алгоритму; $D = \{d_1, d_h\}$, що поступають в автомат як операнди; $R = \{r_1, r_q\}$, що представляє результати операцій; $S = \{s_1, s_n\}$ – внутрішні слова, які представляють інформацію в процесі виконання операцій.

На рисунку 1 показано структуру ОП як комбінацію операційної та керуючої частин і зв'язків між ними.

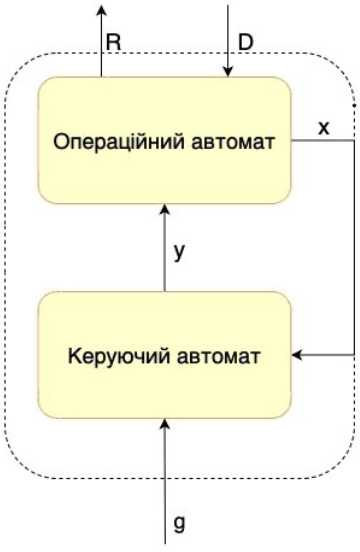

Рисунок 1. Представлення цифрового операційного пристрою як комбінації операційного та керуючого автоматів

Однією з канонічних форм представлення операційного та керуючих блоків є кінцевий автомат (Finite State Machine). Така форма представлення пристрою дозволяє синтезувати дискретні схеми без урахування часових параметрів і конкретних фізичних елементів, з яких ці схеми побудовані. На рисунку 2 наведено приклад графа переходів кінцевого автомата Мура.

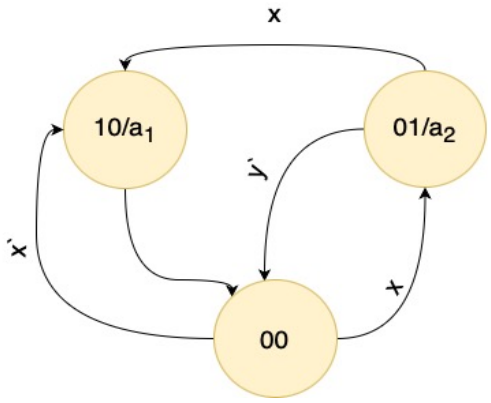

Рисунок 2. Граф переходів кінцевого автомата Мура

У цьому графі дуги означають переходи між станами та умови, за яких вони відбуваються; вершини – стани та вихідні сигнали, які видає автомат при перебуванні у цьому стані. Зокрема в [2] визначено роль і місце керуючих автоматів у системах логічного управління. Подана методика проєктування автоматичних систем логічного управління з урахуванням реального часу и обробки зовнішніх подій. Наведена класифікація зовнішніх подій і способи їх обробки.

Концепція автомата реального часу – це спосіб опису технічних систем реального часу [3, 4]. Графова модель автомата доповнюється скінченною множиною таймерів, які приймають значення натуральних величин. Як і в класичному графі, вершини називаються станами, дуги – переходами. Кожний таймер скидається в нуль у момент переходу і збільшує своє значення з кожним автоматним тактом. З кожним переходом пов'язане часове обмеження (clock constraint), яке означає, що цей перехід може бути здійснений лише в тому випадку, якщо поточні значення таймера задовольняють це обмеження. З кожною позицією пов'язане обмеження на таймери – інваріанти; система може перебувати в цій позиції доти, доки виконується її інваріант.

DOI: 10.24025/2306-4412.2.2023.274840

У [5] введено нову класифікацію моделей кінцевих автоматів залежно від реалізації переходів та вихідних сигналів. Всі автомати розділено на три класи. У автоматів першої категорії (регулярні, regular) переходи залежать виключно від вхідних сигналів, а значення вихідних сигналів – тільки від станів та вхідних сигналів. Це класичні автомати Мілі та Мура. У автоматів другої категорії (часові, timed) переходи залежать від вхідних сигналів та часу їх прояву; вихідні сигнали – тільки від станів. Переходи автоматів третьої категорії (рекурсивні, recursive) залежать від вхідних сигналів та часу їхньої появи, а вихідні – від теперішнього стану та попереднього, тобто для вихідних сигналів y_i у стані a_i реалізується функція $y_i = y_i + y_j$, де a_j попередній стан автомата.

В [6] розглянуто узагальнену модель часового автомата з таймаутами, часовими обмеженнями та вихідними затримками. Узагальнено модель структурного темпорального керуючого автомата $Y(t) = g(X(t), Z(t), T)$, $Z(t+1) = f(X(t), Z(t), T)$, де X – множина вхідних сигналів, Z – множина внутрішніх змінних, яка визначає стан автомата, Y – множина вихідних сигналів, t – машинний час, що визначається в автоматичних тактах, d – функція виходів і f – функція переходів структурного автомата. $T = \{t_c, t_o, t_d\}$ є множиною часових параметрів автомата, де: t_c – часові обмеження, t_o – вхідні таймаути, t_d – вихідні затримки.

Способи опису моделей кінцевих автоматів мовами опису апаратури на основі автоматних шаблонів розглянуто в [7, 8], а в [9, 10] розглянуто проблеми опису та мінімізації моделей часових подієвих автоматів. В [11] розглядаються питання знаходження та аналізу довжини встановлюючих послідовностей для часових автоматів. Моделі параметричних часових автоматів з умовною довжиною вхідних сигналів та інваріантів розглядаються в [12]. В [13] запропоновано спосіб моделювання зовнішніх та внутрішніх дискретних подій з використанням автоматних моделей.

У [14] запропоновано класифікацію керуючих автоматів за способом отримання вихідних сигналів на моделі Мілі та Мура, за способом опрацювання вхідних сигналів на пасивні та активні. Також запропоновано класифікувати події згідно зі способом опрацювання на ініціюючі та перериваючі.

Варто відзначити, що в усіх розглянутих роботах не розглядаються події довжиною більше одного автоматного такту. Проте в технічних системах реального часу виникає задача розгляду моделей часових подієвих автоматів, які обробляють події з невизначеною довжиною. Для спрощення викладення приймемо, що $t_c = t_o$, тобто подія «приймається» та обробляється протягом усього часу знаходження автомата у відповідному стані.

Мета та задачі дослідження. Метою статті є розробка моделей нового класу подій з невизначеною тривалістю для моделювання пристроїв логічного керування реального часу, наданих мовами опису апаратури.

Задачами дослідження є:

– визначення класу пристроїв, в яких зовнішні події мають тривати певний час для впливу на систему;

– формулювання способів вираження такого роду подій на графі переходів;

– формування теоретичного підґрунтя для моделювання нового класу пристроїв;

– моделювання можливих сценаріїв обробки подій мовами опису апаратури.

Об’єктом дослідження є процеси автоматизованого проектування цифрових пристроїв логічного керування реального часу. Предметом дослідження є моделі подій та способи їх обробки в автоматних пристроях керування реального часу, представлених мовами опису апаратури.

Виклад основного матеріалу. Візуальним описом моделі часового автомата є темпоральний граф переходів. Такий граф розширюється таймером, який використовується для затримок у станах. Таймер використовується для перебування у стані впродовж певної кількості тактів синхросигнала. На рисунку 3 наведено приклад автомата з часовими переходами.

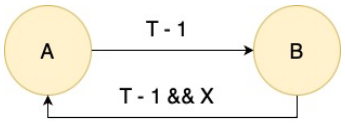

Рисунок 3. Приклад темпорального графа переходів

Перехід між станами A та B – це звичайний часовий перехід, при якому автомат залишається в стані A протягом T циклів син-

DOI: 10.24025/2306-4412.2.2023.274840

хросигнала. Умова написана у формі *T - 1*, оскільки внутрішній таймер прийматиме значення від 0 до *t = T - 1*. Перехід *BA* залежить від стану допоміжного таймера і вхідного сигналу X. Таким чином, автомат перебуватиме у стані B принаймні протягом *T - 1* тактів. Такий тип переходів називається умовно-часовим.

На рисунку 4 детально описані як часові, так і умовні переходи для автомата з рисунка 2 з використанням явного значення таймера в описі умов.

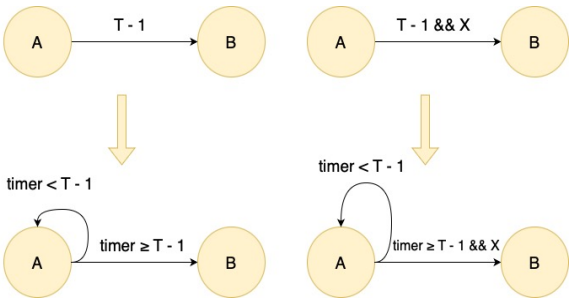

Рисунок 4. Приклади опису темпоральних переходів

Як було зазначено вище, існує клас пристроїв, які реагують на зовнішні події, подовжені у часі. Для таких пристроїв є важливими часові характеристики зовнішньої події. Наприклад, затискання кнопки вимикає пристрій; при зберіганні певного рівня температури у приміщенні впродовж 30 хв має увімкнутися система охолодження; якщо оператор не реагує на помилки у системі впродовж 5 хв, система має увійти в аварійний стан. Цей клас пристроїв вимагає відображення в існуючих моделях на базі кінцевих автоматів.

Визначимо, що часові обмеження події – це значення $t_c(X(i)) = [c1]$, що позначає мінімальний час тривалості певного вхідного сигналу події. У прикладі з кнопкою блокування мобільного телефона $t_c = 3$ *с (для спрощення викладення це значення наведено у секундах)*. Зазначене обмеження виразимо на темпоральному графі у вигляді переходу. В умові цього переходу необхідно враховувати значення таймера зі значенням $t_c(X(i))$.

На рисунку 5 наведено приклад такого переходу. Як і у випадку темпорального переходу, в умові береться до уваги стан таймера. Головною відмінністю є те, що таймер тепер має нижню границю, тобто для успішної зміни стану a_2/a_3 необхідно, аби сигнал події *evnt* був активним (приймав значення високого рівня, тобто ‘1’) і таймер був більшим за значення $t_c(evnt)$. Введемо два логічні підстани стану a_2: *a2/reset* та *a2/inc*. У разі, якщо сигнал *evnt* приймає значення високого рівня, але значення таймера менше за $t_c(evnt)$, то автомат переходить у підстан *a2/inc*. У цьому стані таймер інкрементує внутрішній стан. Якщо сигнал *evnt* приймає значення низького рівня (‘0’) та значення таймера менше за $t_c(evnt)$ – автомат опиняється у підстані *a2/reset*, в якому таймер скидається у 0. Важливо зауважити, що необхідно мати стан, перехід в який не буде залежати від таймера. Це є важливим для уникнення циклів. У цьому прикладі таким станом є a_1, в який автомат переходить при високому рівні сигналу скидання *rst*.

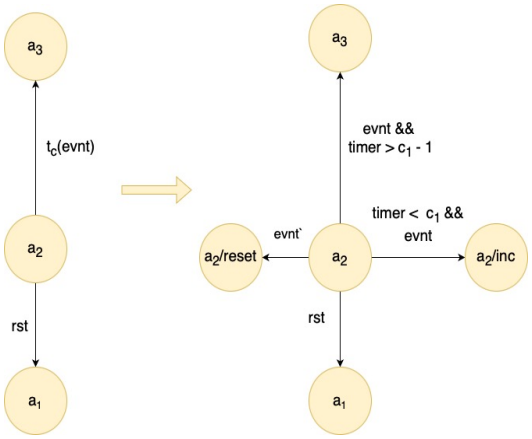

Рисунок 5. Приклад темпорального графа переходів з подією

Залежно від природи зовнішньої події, можуть бути чотири можливі типи поведінки, які задовольняють або порушують часову вимогу. Для подальшого викладення приймемо, що сигнал *evnt* має зберігати високий логічний рівень впродовж мінімум 4 тактів сигналу *Clk*. Перший сценарій – це коли зовнішня подія триває рівно необхідний час. У цьому випадку автомат переходить у наступний стан і внутрішній лічильник скидається. На рисунку 6 показано часову діаграму – сигнал *evnt* зберігає високе значення впродовж 4 тактів синхросигналу *Clk (*мінімально необхідна тривалість для зміни станів*)*, після чого автомат переходить у новий стан *a3*. Червоним пунктиром позначено початок та кінець обрахування події внутрішнім таймером.

DOI: 10.24025/2306-4412.2.2023.274840

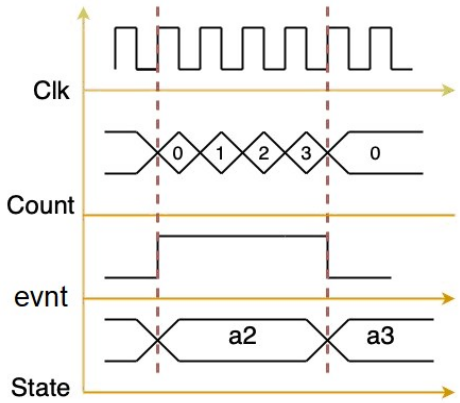

Рисунок 6. Часова діаграма темпорального переходу з подією, яка триває рівно необхідну кількість тактів

Другий випадок відповідає ситуації, коли подія триває більше, ніж t_o відповідного стану. Варто відзначити, що для класу пристроїв, описаних у вступі, більша тривалість події не є критичною, оскільки виконується умова мінімальної тривалості події. Як і в попередньому випадку, автомат перейде в наступний стан зі скиданням лічильника. Рисунок 7 містить часову діаграму такого випадку.

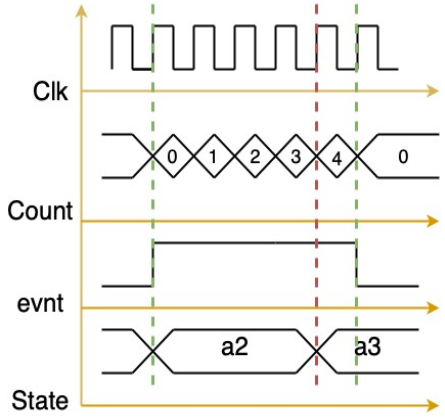

Рисунок 7. Часова діаграма з подією, тривалістю більше, ніж необхідно

Як видно з діаграми, сигнал *evnt* зберігає високе значення впродовж 5 тактів, що є на 1 такт більше від необхідного значення. Як і в минулому випадку, автомат перейде у новий стан та перезапустить таймер. Зеленим пунктиром виділено відрізок, на якому сигнал *evnt* має позитивне значення, червоним – момент зміни стану автомата.

Третій випадок описує ситуацію, коли подія триває менше зазначеного часу. У цьому випадку автомат залишається в тому самому стані зі скиданням лічильника, як показано на рисунку 8. Скидання лічильника у цьому випадку є важливою операцією, оскільки таймер має бути готовий до обробки наступних потенційних подій.

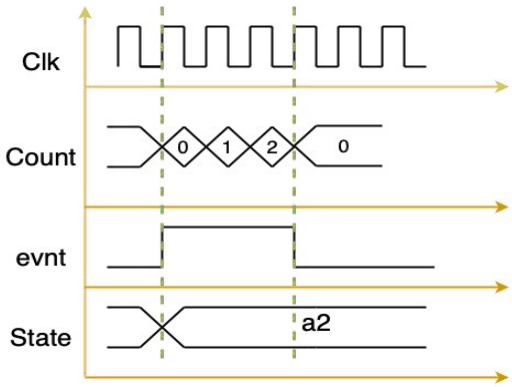

Рисунок 8. Часова діаграма з подією, тривалістю менше, ніж необхідно

Важливим є питання тупика, в який може потрапити автомат, якщо зовнішня подія не триває необхідний час або не відбувається взагалі. Для такого випадку автомат повинен мати сигнал скидання, який переводив би пристрій у наперед заданий стан. Як правило, для цього інтерфейс цифрового пристрою має сигнал скидання. Важливо відзначити, що цей сигнал має вищий пріоритет, ніж інші вхідні сигнали. Це є важливим під час написання безпосередньо апаратної реалізації пристрою за допомогою мов опису апаратури, таких як VHDL, Verilog, SystemVerilog. Рисунок 9 містить часову діаграму, яка описує цю ситуацію – сигнал *evnt* зберігає (приймає) високий рівень одночасно з сигналом *rst*, однак автомат переходить у стан a_1, оскільки сигнал *rst* має більший пріоритет.

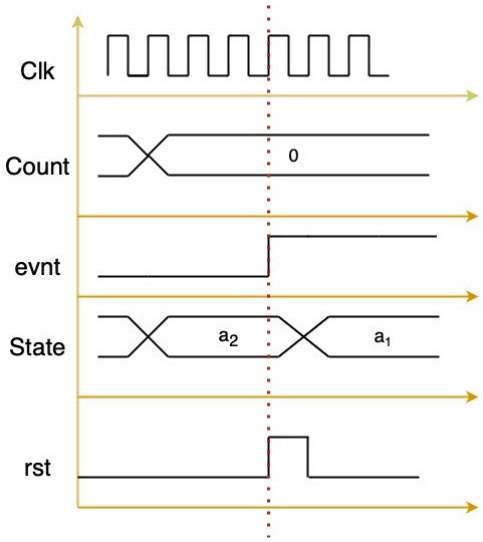

Рисунок 9. Часова діаграма темпорального переходу при асинхронному скиданні

DOI: 10.24025/2306-4412.2.2023.274840

Результати дослідження. Розділення пристроїв на операційний та керуючий автомати є найпоширенішою формою проектування цифрових пристроїв. Ця форма візуально виражає внутрішню природу пристрою на високому рівні абстракції. На базі графа переходів команда проектувальників описує пристрій за допомогою мови опису апаратури, використовуючи концепцію автоматного шаблону.

Темпоральний граф переходів є відправною точкою при проєктуванні керуючої частини пристроїв реального часу. Така модель графа розширюється дискретною множиною таймерів, які використовуються для опису затримок у станах. Обробка подій в такого роду моделях теж має свої особливості.

Запропоновано новий клас пристроїв, у яких тривалість зовнішніх подій має задовольняти часовим умовам, аби система на них реагувала. Ці обмеження необхідно враховувати при проектуванні кінцевого керуючого автомата, а відповідно, і відображати на темпоральному графі переходів. У роботі запропоновано новий вид переходу, який враховує не тільки зовнішню подію, але й її тривалість. Технічно це реалізується за допомогою додаткового таймера.

Для запропонованої моделі розглянуто різні випадки тривалості події та відповідні сценарії реакції автомата на них:

– зовнішня подія триває рівно стільки часу, скільки вказано у специфікації;

– зовнішня подія триває менше необхідного часу;

– зовнішня подія триває більше необхідного часу;

– зовнішня подія не відбувається.

Усі можливі сценарії проілюстровано відповідними часовими діаграмами з детальними поясненнями та кодом мовою опису апаратури. На рисунку 10 наведено опис фрагменту моделі керуючого пристрою мовою опису апаратури (в цьому випадку використано Verilog, підмножина, що синтезується), який ілюструє часову діаграму на рисунку 9.

Сигнал *rst_n* (скидання) обробляється перед рештою сигналів, тобто має найвищий пріоритет. Цей фрагмент виділено червоним пунктиром, обробку інших сигналів (логіку зміни станів) – зеленим.

```
always @(posedge clk, posedge rst_n)
  if (rst_n) begin
      state <= A;
      timer <= 3'd0;
  end
  else begin
      state <= next_state;
      timer <= next_timer;
  end

always @(state, timer, x) begin
  next_state = state;
  next_timer = 3'd0;
  case (state)
    A: if (timer >= Tmax) begin
          next_timer = 3'd0;
          next_state = B;
    end
    else begin
          next_state = A;
          next_timer = timer + 1'b1;
    end
    B: next_state = C;
    C: if (timer >= Tmax && x) begin
          next_state = A;
          next_timer = 3'd0;
       end
          else begin
               next_state = B;
               next_timer = timer + 1'b1;
          end
    default: begin
     next_state = A;
     next_timer = 3'd0;
    end
  endcase
end
```

Рисунок 10. Приклад автоматного опису мовою Verilog з обробкою подій

Обговорення результатів. Серед сучасних технічних систем логічного управління та зв'язку все більшу роль відіграють пристрої з сенсорними панелями керування. Для таких пристроїв суттєвий вплив має не тільки наявність керуючих подій, але і їх тривалість. З другого боку, людина, задаючи керуючу подію (натискаючи на сенсорну панель), не має можливості точно регулювати тривалість сенсорного контакту (події). Виходячи з цього, пристрій керування (керуючий автомат) повинен мати змогу коректно реагувати на керуючі зовнішні події невизначеної тривалості. В роботі розглянуто різні співвідношення тривалості подій та станів керуючого часового автомата. Запропоновано способи обробки зовнішніх подій у системах логічного управління реального часу, реалізовані в моделях мовами опису апаратури, що дозволяє використовувати зазначені процедури в системах автоматизованого проєктування керуючих пристроїв на основі систем програмованої логіки.

Висновки. У роботі розглянуто питання проєктування пристроїв реального часу на базі графів переходів керуючих автоматів. Детально розглянуто новий клас пристроїв реального часу – пристрої, які функціонують

DOI: 10.24025/2306-4412.2.2023.274840

на базі подій з певною тривалістю у часі. Такий рід подій було запропоновано відображати на темпоральному графі переходів. Детально розглянуто можливі реакції автомата залежно від тривалості події.

Наукова новизна роботи полягає в удосконаленні темпорального графа переходів як основної моделі візуального відображення пристроїв логічного керування реального часу, що дозволило значно розширити клас подієвих пристроїв логічного керування, які можуть бути представлені автоматними моделями.

Практична цінність роботи полягає в наступному:

– виокремлено новий клас пристроїв логічного керування реального часу, що враховують наявність зовнішніх подій, і відповідний спосіб їхнього представлення мовами опису апаратури та моделювання на базі кінцевого автомата;

– на прикладах показано можливість використання запропонованої моделі подій при різних співвідношеннях тривалості подій та часу знаходження керуючого часового автомата у відповідному стані.

Запропоновані моделі є теоретичною базою для удосконалення процесу автоматизованого проектування керуючих автоматів за допомогою мов опису апаратури з використанням автоматного шаблону.

Напрямком подальших досліджень може бути використання отриманих результатів у розробці систем діагностики подієвих систем реального часу [15].

Список використаних джерел / References

[1] S. Baranov, *Logic and System Design of Digital Systems*. Tallinn: TUT Press, 2008.

[2] A. A. Shalyto, "Software automation design: Algorithmization and programming of problems of logical control", *Journal of Computer and System Sciences International*, vol. 39, no. 6, pp. 899-916, 2000.

[3] R. A. Alur, and D. L. Dill, "Theory of timed automata", *Theoretical Computer Science*, vol. 126, no. 2, pp. 183-235, 1994.

[4] M. Zhigulin, N. Yevtushenko, S. Maag and A. Cavalli, "FSM-based test derivation strategies for systems with time-outs", in *2011 11th Int. Conf. Qual. Softw. (QSIC)*, Madrid, Spain, July, 13–14, 2011. [Online]. Available: https://doi.org/10.1109/qsic.2011.30. Accessed on: March 29, 2023.

[5] V. A. Pedroni, *Finite State Machines in Hardware: Theory and Design (with VHDL and SystemVerilog)*. MA: MIT Press Cambridge, 2013.

[6] M. Miroshnyk et al., "Design of real-time system logic control on FPGA", in *2019 IEEE East-West Des. Test Symp. (EWDTS)*, Batumi, Georgia, Sept. 13-16, 2019. [Online]. Available: https://doi.org/10.1109/ewdts.2019.8884387. Accessed on: March 29, 2023.

[7] A. S. Klimowicz, and V. V. Solov'ev, "Structural models of finite-state machines for their implementation on programmable logic devices and systems on chip", *J. Comput. Syst. Sci. Int.*, vol. 54, no. 2, pp. 230-242, March 2015. [Online]. Available: https://doi.org/10.1134/s1064230715010074. Accessed on: March 29, 2023.

[8] M. A. Miroshnyk, A. S. Shkil, E. N. Kulak, D. Y. Rakhlis, A. M. Miroshnyk, and N. V. Malahov, "Design timed FSM with VHDL Moore pattern", *Radio Electronics, Comput. Science, Control*, no. 2, pp. 137-148, Sept. 2020. [Online]. Available: https://doi.org/10.15588/1607-3274-2020-2-14. Accessed on: March 29, 2023.

[9] D. Bresolin, A. Tvardovskii, N. Yevtushenko, T. Villa, M. Gromov, "Minimizing deterministic timed finite state machines", in *14th IFAC Workshop on Discrete Event Systems WODES 2018,* IFAC-PapersOnLine, vol. 51, issue 7, pp. 486-492, 2018.

[10] D. Bresolin, K. El-Fakih, T. Villa, and N. Yevtushenko, "Equivalence checking and intersection of deterministic timed finite state machines", *Formal Methods in System Design*, no. 7, pp. 1-26, 2022.

[11] A. S. Tvardovskii, and N. V. Yevtushenko, "Deriving homing sequences for finite state machines with timed guard", *Automatic Control and Computer Sciences*, vol. 55, no. 7, pp. 738-750, 2021.

[12] E. André, D. Lime, and M. Ramparison "TCTL model checking lower/upper-bound parametric timed automata without invariants", in *Proc. Int. Conf. Formal Modeling and Analysis of Timed Systems FORMATS 2018*, Sept. 4-6, Biejing, China, 2018, pp. 37-52.

[13] G. Wagner, "An abstract state machine semantics for discrete event simulation", in *Proc. 2017 Winter Simulation Conf. (WSC)*, Dec. 3-6, 2017, Las Vegas, USA. [Online]. Available: https://ieeexplore.ieee.org/ document/8247830. Accessed on: March 29, 2023.

DOI: 10.24025/2306-4412.2.2023.274840

[14] M. Miroschnyk, A. Shkil, E. Kulak, D. Rakhlis, I. Filippenko, and M. Malakhov, "Hardware implementation of timed logical control FSM", *Proc. 2020 IEEE East-West Design & Test Symposium (EWDTS'20)*, Sept. 4-7, Varna, Bulgaria, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9225129. Accessed on: March 29, 2023.

[15] Lamperti, and M. Zanella, *Diagnosis of Active Systems: Principles and Techniques (The Springer International Series in Engineering and Computer Science).* Springer, 2003.

M. A. Miroshnyk[1], *Dr. Tech. Sc., Professor,*
e-mail: m.miroshnyk@karazin.ua
O. S. Shkil[2], *Ph. D., Associate Professor,*
e-mail: oleksandr.shkil@nure.ua
D. Yu. Rakhlis[2], *Ph. D., Associate Professor,*
e-mail: dariia.rakhlis@nure.ua
K. Yu. Pshenychnyi[2], *Ph. D. Student,*
e-mail: kyrylo.pshenychnyi@nure.ua
A. M. Miroshnyk[2], *Assistant,*
e-mail: anatolii.miroshnyk@nure.ua
[1]Kharkiv National University named after V. N. Karazin
Svobody Maidan, 4, Kharkiv, 61000, Ukraine
[2]Kharkiv National University of Radioelectronics
Nauky Ave. 14, Kharkiv, 61166, Ukraine

EVENT PROCESSING MODEL FOR SIMULATION OF REAL-TIME LOGIC CONTROL DEVICES

Among modern technical systems, logical control and communication devices with touch control panels play an increasingly important role. In such devices the duration of controlling events is important. On the other hand, a person cannot exactly control the durability of the event when pressing and holding a panel. Based on this, a control device (control unit) must properly respond to external events of unknown durability. Thus, when designing a real-time system, it's required to express metric time in terms of the automation clock because the transitions between the automata states directly depend on the metric time aspect.

The purpose of this article is to introduce a new events class used in real-time device simulation – the events with minimal duration. The object of research is real-time device control algorithms. The subject of research is event-based automatic models described by the hardware description languages.

The article analyzes the issue of simulation real-time devices using the state machine template in hardware description languages. The classification of events as finite state machine interaction with the external environment models is analyzed. A new type of event and a class of devices whose functionality depends on such factors are introduced. It is shown that such events are widespread in various digital devices and real-time systems. A new type of FSM transition is proposed for the temporal state diagram as digital systems design canonical model. The considered models are illustrated by simulations and timing diagrams analysis.

For the proposed model, different cases of the duration of the event have been considered and the corresponding processes of the machine's reaction to them are as follows: the external event lasts exactly as long as required in the specification; the external event lasts less than required; the external event lasts longer than required; no external event occurs. All possible processes are illustrated with timing diagrams with detailed explanations and hardware description language code examples. A description of the device model fragment using the Verilog synthesizable subset is given.

Keywords: *digital devices, simulation, real-time models, signals, events, language models of timed finite state machines.*

DOI: 10.24025/2306-4412.2.2023.274840