|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

Институт искусственного интеллекта

Кафедра проблем управления

**ПРАКТИЧЕСКАЯ РАБОТА №2**

по дисциплине **Операционные системы реального времени**

**Тема практической работы: «**Исследование процесса перехода контроллера в сервисный

режим и влияние параметра tolerance на выполнение задач»

|  |  |
| --- | --- |
| **Студенты группы:** КРБО-03-23 | Зенина А. А. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
|  | Грачев А. А. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
|  | Гришаев А. К. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Преподаватель:** | ст. преподаватель Смирнов М.Ю. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

|  |  |
| --- | --- |
| Работа представлена к защите: | «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2025 г. |

Москва 2025

# **1. Цель работы**

Изучить работу программ в сервисном режиме контроллера, а также влияние параметра tolerance на выполнение задач.

# **2. Задание**

1. В среде Automation Studio создать программу, переводящую контроллер в сервисный режим.

2. Определить причину перехода контроллера в сервисный режим с помощью встроенных средств диагностики.

3. В настройках конфигурации активировать exception task class.

4. Выставить нулевой допуск по времени и снять показания профилировщика.

5. Подобрать необходимый допуск по времени, который позволил бы программе завершиться.

# **3. Теоретические сведения**

Automation Runtime - операционная система реального времени для промышленных контроллеров. Процесс начальной загрузки может быть разделен на четыре этапа:

1. BOOT (НАЧАЛЬНАЯ ЗАГРУЗКА) - возникает если не вставлена карта памяти CompactFlash, на карте CF нет операционной системы или переключатель узла выставлен на "00"

2. DIAG (ДИАГНОСТИКА) - возникает при очистке памяти, неустранимой системной ошибке или переключателе узла выставленном на "FF"

3. SERV (СЕРВИС) - возникает при делении на ноль, ошибке доступа к памяти (Page fault), превышении времени цикла, остановке ЦПУ по команде из Automation Studio или прочих ошибках

4. RUN (РАБОТА) - нормальный режим работы без ошибок

Для каждого класса задач предусмотрено заданное время цикла. Все задачи в своем классе должны быть выполнены в течение этого времени цикла. Если суммарное время выполнения задач превышает настроенное время цикла, выдается ошибка превышения времени цикла (cycle time violation). Параметр tolerance (допустимое отклонение) предотвращает переход системы в сервисный режим при незначительном превышении времени цикла.

# **4. Основная часть**

## 4.1. Описание действий и использованного аппарата

Лабораторная работа выполнялась на контроллере B&R с использованием среды разработки Automation Studio. Для перевода контроллера в сервисный режим была создана программа с долгим циклом:

for(int i = 0; i < 9974970; i = i + 1){  
 counter = counter + 1;  
}

Данный код приводит к превышению времени выполнения задачи и переводу контроллера в сервисный режим.

Структурная схема лабораторного стенда включает:

- Контроллер B&R с установленной Automation Runtime

- Рабочую станцию с Automation Studio

- Сетевое соединение между ПК и контроллером

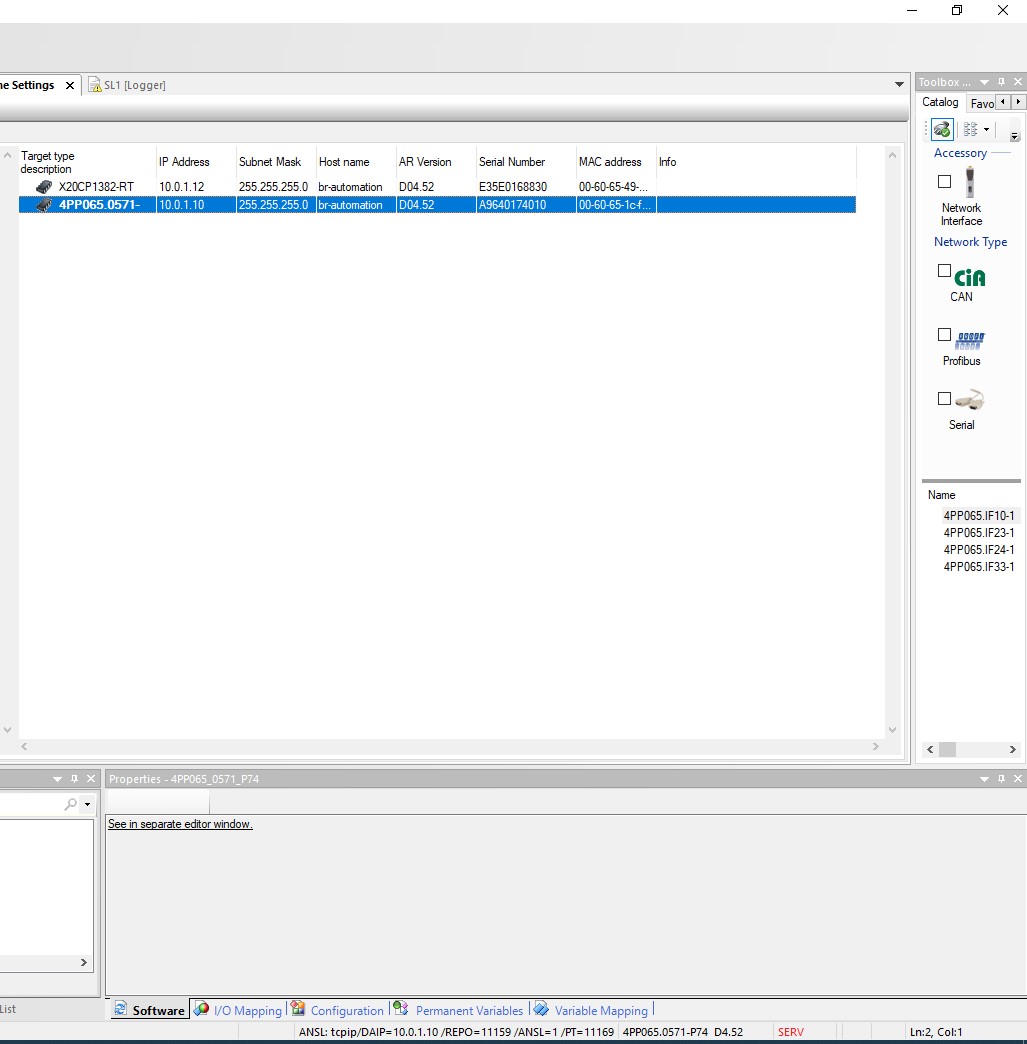
Для диагностики использовались встроенные средства Automation Studio:

- Окно Online Settings для мониторинга состояния контроллера

- Профилировщик задач для анализа времени выполнения

- Диалоговые окна конфигурации параметров tolerance

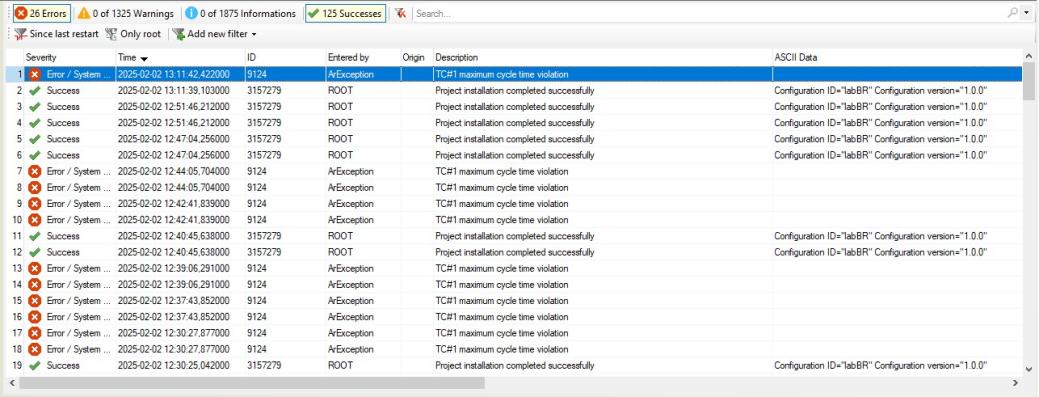
На Рисунке 1 показан контроллер в сервисном режиме после превышения времени выполнения задачи.



*Рисунок 1 - Контроллер в сервисном режиме*

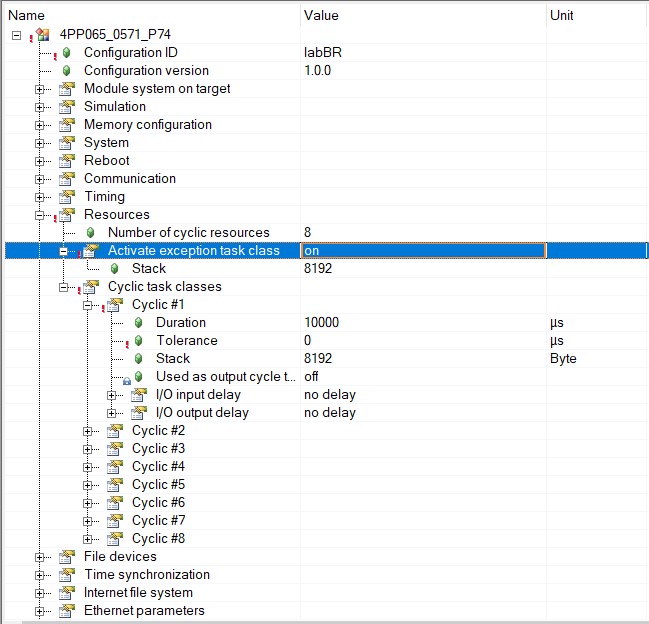
## 4.2. Результаты экспериментальных исследований

При нулевом допуске по времени (tolerance = 0) программа не может завершиться, так как время выполнения значительно превышает установленный цикл. На Рисунке 2 видна ошибка превышения максимального времени выполнения задачи.



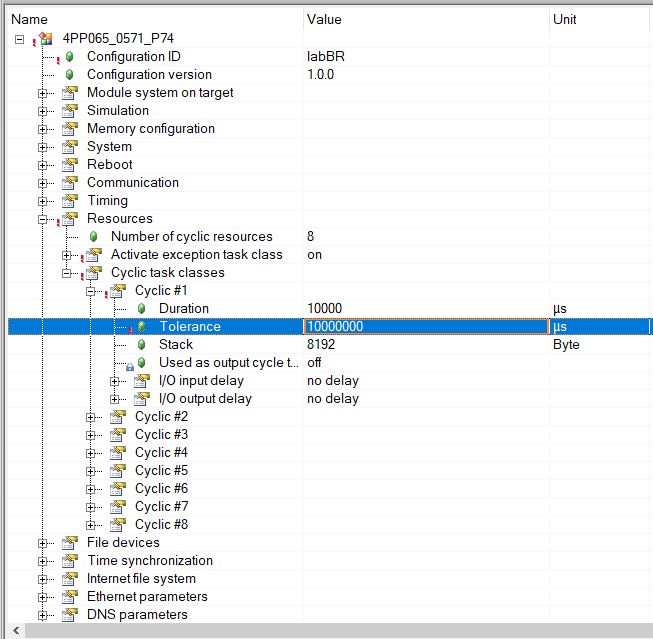
*Рисунок 2 - Ошибка превышения времени выполнения задачи*

После активации exception task class (Рисунок 3) система получает возможность обработки исключений без перехода в сервисный режим.



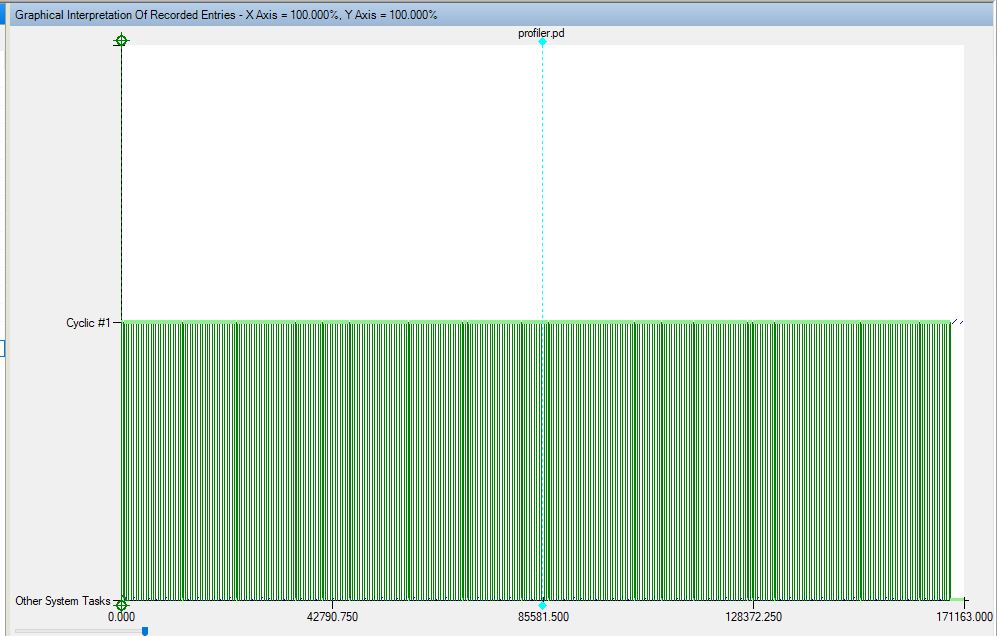
*Рисунок 3 - Активирован режим exception task class*

Подбор оптимального значения tolerance показал, что при установке соответствующего допуска (Рисунок 4) программа успешно завершается без перехода в сервисный режим.



*Рисунок 4 - Установка нового значения tolerance*

На Рисунке 5 демонстрируется успешное выполнение программы после настройки параметра tolerance.



*Рисунок 5 - Успешное выполнение программы после настройки tolerance*

# **5. Выводы по работе**

В ходе практической работы были изучены процессы перехода контроллера в сервисный режим и влияние параметра tolerance на выполнение задач. Экспериментально подтверждено, что:

1. Превышение времени выполнения задачи приводит к переходу контроллера в сервисный режим

2. Параметр tolerance позволяет задать допустимое отклонение по времени выполнения

3. Активация exception task class обеспечивает обработку исключений без перехода в сервисный режим

4. Правильная настройка tolerance позволяет программам завершаться успешно даже при незначительном превышении времени цикла

Цель работы достигнута - изучены механизмы работы программ в сервисном режиме и влияние параметра tolerance на выполнение задач.

# ПРИЛОЖЕНИЕ 1

Листинг программы, приводящей к переходу в сервисный режим:

#include <bur/plctypes.h>  
#ifdef \_DEFAULT\_INCLUDES  
#include <AsDefault.h>  
#endif  
// amount of memory to be allocated for heap storage must be specified for every ANSI C++ program with the bur\_heap\_size variable  
unsigned long bur\_heap\_size = 0xFFFF;  
void \_INIT ProgramInit(void)  
{  
}  
void \_CYCLIC ProgramCyclic(void)  
{  
 for(int i = 0; i < 9974970; i = i + 1){  
 counter = counter + 1;  
 }  
}  
void \_EXIT ProgramExit(void)  
{  
}