**Sprawozdanie do projektu z Systemów Wbudowanych.**

**Autorzy:**

**Grzegorz Listwan**

**Krzysztof Pacura**

Wykonany projekt obejmuje sterowanie oświetleniem w salach na podstawie liczby osób w pomieszczeniach. System posiada czujniki dwukierunkowe na wejściach które odpowiadają za wejście lub wyjście osoby z sali sąsiedniej. Nasz projekt opiera się na poniższym schemacie.

Obraz zawierający diagram, tekst, linia, numer

Opis wygenerowany automatycznie

System został napisany przy użyciu biblioteki SystemC. Do realizacji wejść zostały użyte klawisze numeryczne 0-9 dla wejść CD1-CD10 oraz klawisz „p” dla wejścia CD11 ponadto użyto klawisza „k” do zmiany kierunku na wejściach i klawisza „q” do przerwania symulacji. Stan czujników S wyświetlany jest na ekranie. Założenie jest takie że jeśli wystąpi błąd to jest on kasowany po upływie 0,5s.

**Poniżej szczegółowy opis zadania:**

Projekt obejmuje implementację systemu zarządzania oświetleniem w pomieszczeniach przy użyciu SystemC. Główne moduły to ***Channel,*** ***CPU1***, ***CPU2,*** ***HardwareTask*** oraz ***Room.*** Kanał komunikacyjny (Channel) obsługuje wymianę informacji o liczbie osób w poszczególnych pomieszczeniach. Moduły **CPU1** i **CPU2** kontrolują różne pokoje poprzez zadania (`ask), które aktualizują stan oświetlenia na podstawie liczby osób otrzymanej z kanału. **HardwareTask** reprezentuje zadania sprzętowe. Główna funkcja (main.cpp) inicjalizuje te moduły i rozpoczyna symulację SystemC.

**Kanał Komunikacyjny**

Kanał komunikacyjny (Channel) jest kluczowym komponentem systemu, który umożliwia wymianę informacji między różnymi modułami. W naszym projekcie, Channel przechowuje liczbę osób w poszczególnych pokojach. Funkcje send i receive pozwalają modułom na zapis i odczyt tych danych. Synchronizacja dostępu do danych odbywa się przy użyciu mutexu (sc\_mutex), co zapewnia bezpieczeństwo operacji w środowisku wielowątkowym.

**Zadania Realizowane Sprzętowo**

Moduł HardwareTask symuluje zadania realizowane sprzętowo. Jest to reprezentacja procesów, które w rzeczywistych systemach byłyby realizowane przez dedykowane urządzenia lub układy elektroniczne. Zadanie sprzętowe (task) w naszym projekcie działa w pętli, wykonując określone operacje w regularnych odstępach czasu. To zadanie może być wykorzystywane do monitorowania stanu systemu lub wykonywania innych zadań niezależnych od głównych procesów.

**Wątki SystemC**

Wątki SystemC (SC\_THREAD) są podstawowym mechanizmem do realizacji współbieżności w modelach SystemC. Każdy wątek reprezentuje niezależną sekwencję operacji, które mogą być wykonywane równolegle. W naszym projekcie, moduły CPU1 i CPU2 zawierają wątki, które cyklicznie sprawdzają stan liczby osób w przypisanych pokojach i odpowiednio kontrolują oświetlenie. Wątki te korzystają z funkcji wait, aby wstrzymać swoje działanie na określony czas, co pozwala na symulowanie opóźnień i cykliczności operacji w systemie.

**Podsumowanie**

W projekcie wykorzystano:

- SystemC do modelowania systemu, co umożliwia symulację i analizę zachowań systemów wbudowanych.

- Mutex do synchronizacji dostępu do zasobów współdzielonych, zapewniając bezpieczeństwo operacji w środowisku wielowątkowym.

- Wątki SystemC do równoczesnego wykonywania zadań, co pozwala na symulację równoległych procesów w systemie.

- Standardowe biblioteki C++ dla dodatkowych funkcji i synchronizacji, takie jak thread i chrono do zarządzania czasem i wątkami na poziomie systemu operacyjnego.

Ten projekt stanowi przykład efektywnego wykorzystania narzędzi i technik modelowania systemów wbudowanych, umożliwiając tworzenie i testowanie skomplikowanych scenariuszy przed ich implementacją w rzeczywistym sprzęcie.