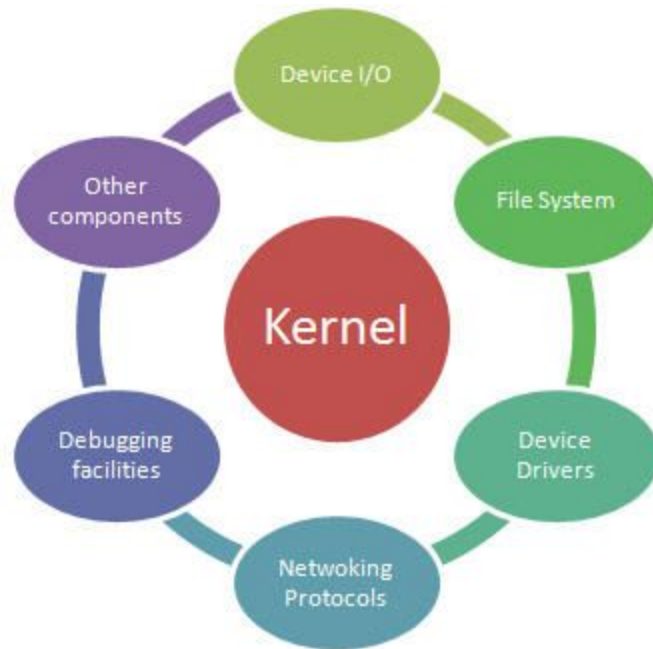


RTOS ONDERZOEK

implementatie concurrency mechanismen door open source RTOS



B1 BAZEN

Team:

Bob Thomas
Robbie Valkenburg
Bart van der Kolk
Jim Bransen

Begeleider:

Jan Zuurbier

INLEIDING

Tijdens een project op school moesten we een laser tag systeem schrijven met gebruik van het Arduino RTOS die beschikt over tasks en de concurrency mechanismen zoals pool, channel, flag(group), clock timer en mutex. Het doel van dit onderzoeksverslag is uit te vinden of er meerdere open source RTOS systemen bestaan die over dezelfde functionaliteit beschikken als de Arduino RTOS. Om deze vraag te kunnen beantwoorden hebben een hoofdvraag opgesteld en daarbij een aantal deelvragen om via onderzoeksresultaten te kunnen aantonen of er meerdere RTOS systemen bestaan zoals de Arduino RTOS en welke de beste vervanger blijkt te zijn.

Hoofdvraag

Met behulp van welke open source real time operating system(rtos) kunnen tasks en de concurrency mechanismen zoals pool, channel, flag(group), clock timer en mutex, zoals aangeboden door het Arduino RTOS, met zo weinig mogelijk overhead worden gerealiseerd?

Deelvragen

1. Wat zijn de kenmerkende eigenschappen van tasks en de concurrency mechanismen van het Arduino RTOS?
2. Welke open source RTOS-en zijn beschikbaar?
3. Welk van de beschikbare RTOS-en biedt de meeste van de concurrency mechanismen van het Arduino RTOS aan zonder enige modificatie en biedt dezelfde functionaliteit om taken te realiseren?
4. Welke mechanismen van het Arduino RTOS worden niet ondersteund door de beschikbare RTOS-en?
5. Hoe kunnen de mechanismen van het Arduino RTOS die niet direct worden ondersteund door de beschikbare RTOS-en worden gerealiseerd m.b.v. van deze RTOS-en?

GEBRUIKTE ONDERZOEKSMETHODEN

De methoden die zijn gevolgd om elk van de deelvragen te beantwoorden.

Deelvraag 1

Wat zijn de kenmerkende eigenschappen van tasks en de concurrency mechanismen van het Arduino RTOS?

Om deelvraag 2 te kunnen beantwoorden gebruiken wij de “bieb” uit de verschillende onderzoeksmethodes. We hebben gekozen voor de “bieb” omdat we literatuurstudie gaan toepassen door gebruik te maken van de documentatie van het Arduino RTOS

Deelvraag 2

Welke open source RTOS-en zijn beschikbaar?

Om deelvraag 2 te kunnen beantwoorden gebruiken wij de “bieb” uit de verschillende onderzoeksmethodes. We hebben gekozen voor de “bieb” methode omdat we verschillende RTOS systemen moeten vergelijken en we alle informatie voornamelijk op het internet kunnen vinden.

Hoe we deelvraag 2 aanpakken is door een lijstje op te stellen van de minimale eisen voor het RTOS systeem.

- Het RTOS moet beschikbaar zijn voor het ARM platform
- Het RTOS moet C en C++ ondersteunen
- Het RTOS moet binnen het afgelopen jaar een update hebben gekregen
- Het RTOS moet beschikken over documentatie waarin de geboden functionaliteit is beschreven.

Met dat lijstje gaan we het internet op en zullen we een inventarisatie opstellen en zullen we een top 10 en die gaan vergelijken tegenover de Arduino RTOS

Deelvraag 3

Welk van de beschikbare RTOS-en biedt de meeste van de concurrency mechanismen van het Arduino RTOS aan zonder enige modificatie en biedt dezelfde functionaliteit om taken te realiseren?

Om deelvraag 3 te kunnen beantwoorden gebruiken wij de “bieb” uit de verschillende onderzoeksmethodes. We hebben gekozen voor de “bieb” omdat we het Arduino RTOS gaan vergelijken tegenover de beschikbare rtos systemen

Deelvraag 4

Welke mechanismen van het Arduino RTOS worden niet ondersteund door de beschikbare RTOS-en?

Deelvraag 5

Hoe kunnen de mechanismen van het Arduino RTOS die niet direct worden ondersteund door de beschikbare RTOS-en worden gerealiseerd m.b.v. van deze RTOS-en?

RESULTATEN

De informatie die is verzameld om het antwoord op elk van de deelvragen te geven met verwijzingen naar bronnen waaruit de informatie is gehaald, volgens APA, (zie (Bronvermelding volgens de richtlijnen van APA) en b.v. (SCRiBBR APA Generator, n.d.). Maak voor overzichten van de verzamelde gegevens gebruik van tabellen.

CONCLUSIES EN AANBEVELINGEN

Let erop dat de conclusies worden onderbouwd op basis van de resultaten van het onderzoek. Begin met het antwoord op de hoofdvraag en geef daarbij aan op welk van de resultaten dit antwoord is gebaseerd. Beantwoord vervolgens elk van de deelvragen en verwijs daarbij naar de resultaten waarop die antwoorden zijn gebaseerd. Bij de aanbevelingen kan b.v. worden beschreven welk onderzoek nog moet worden uitgevoerd om het RTOS, dat als (een van de) beste uit de bus komt, geschikt kan worden gemaakt voor het bieden van de mechanismen van het Arduino RTOS.

DEELVRAAG 1

Wat zijn de kenmerkende eigenschappen van tasks en de concurrency mechanismen van het Arduino RTOS?

Als onderzoeksmethode heb ik lab gekozen omdat met de Arduino RTOS library in de praktijk getest kan worden wat de eigenschappen zijn.

Het Library aspect

Voor het gebruiken van de Arduino RTOS is slechts één library nodig. De library is **91.5 KB** en de header file van de library zal eenmalig in de main moeten worden ge-include. Om een klasse de mogelijkheden van RTOS te bieden, dient de klasse wel een interface te hebben van de RTOS library. De library stelt geen verdere eisen van programma architectuur (geen verdere frameworks nodig).

- 91.5 KB library
- Een interface voor class die Arduino RTOS wilt gebruiken.

Eigenschappen van de Arduino RTOS task

Een eigenschap van de RTOS task is dat het eenvoudig te definiëren is bij een klasse door slechts een void functie genaamd *main* toe te voegen. Bij het compileren van de code zal de Arduino RTOS deze functie gaan herkennen als een task. Bij het gebruiken van de wait functie uit de Arduino RTOS library, zal tijdens het wachten gekeken worden of een andere taak actief is en deze verder mag gaan. Ook is het mogelijk om prioriteiten te geven aan de RTOS task.

Eigenschappen concurrency mechanismen

(I dont know)

