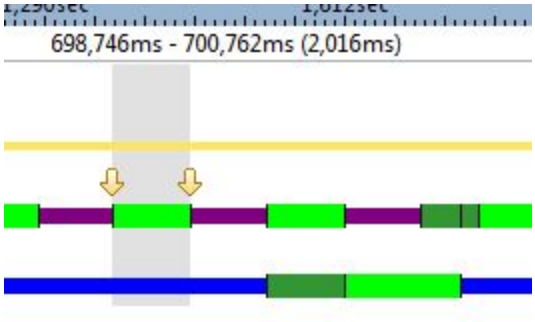
Abgabe Aufgabenblatt Nr. 4

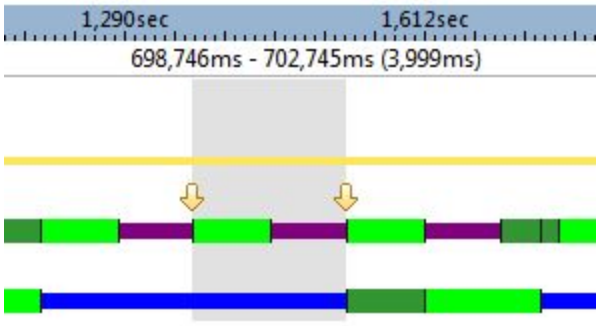
Maximilian Gaul
Markus Schmidt

Task 1: Zykluszeit 4ms – Verarbeitungszeit 2ms

In diesem Abschnitt lässt sich gut erkennen, dass Thread 1 jeweils 2ms (hellgrüner Abschnitt) arbeitet.



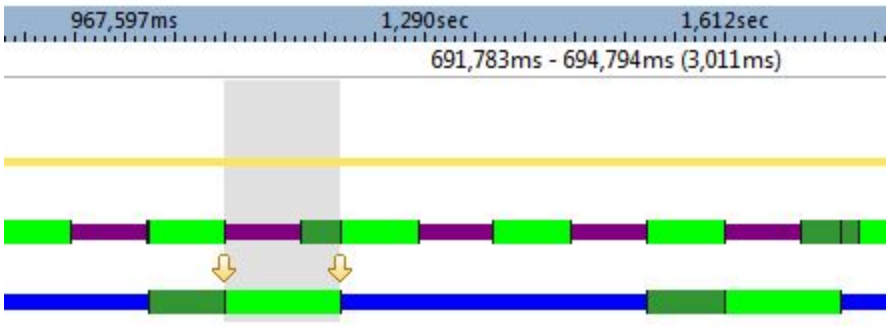
Die Zykluszeit von 4ms setzt sich aus dem Beginn der Verarbeitungszeit und der restlichen Wartezeit zusammen. (Violetter Anteil).
Die Wartezeit wird als absolute Zeit realisiert. Aktuelle Zeit + 4ms.



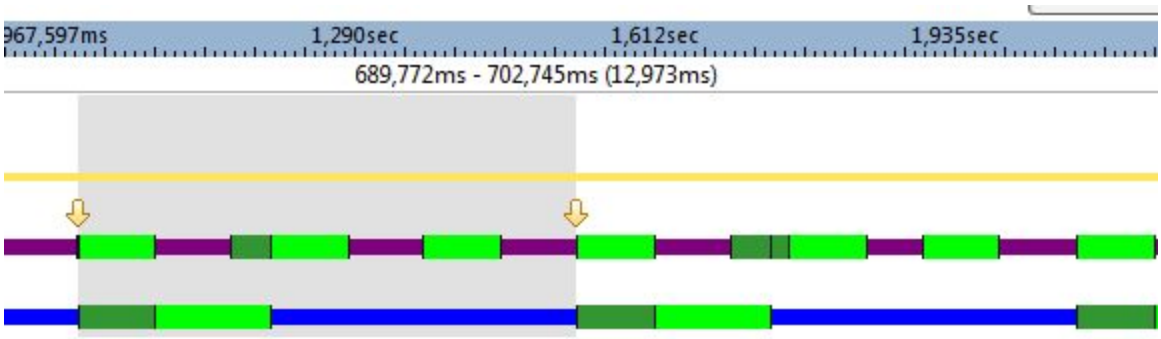
Task 2: Alle 3 Takte wird von Task 1 Semaphore gesetzt

Verarbeitungszeit 3ms

Der hellgrüne Bereich von Thread 2 (untere Timeline) symbolisiert die Verarbeitungszeit von 3ms.
Der dunkelgrüne Bereich vor diesem zeigt an, dass der Thread bereit gewesen wäre zur Ausführung (die Semaphore bereits gesetzt), aber ihm keine Rechenzeit zugewiesen wurde. Thread1 hat noch aktiv den Rechenkern genutzt. Thread 2 muss somit noch so lange warten, bis die Verarbeitungszeit von 2ms von Thread 1 verstrichen sind.



Warten auf Semaphore

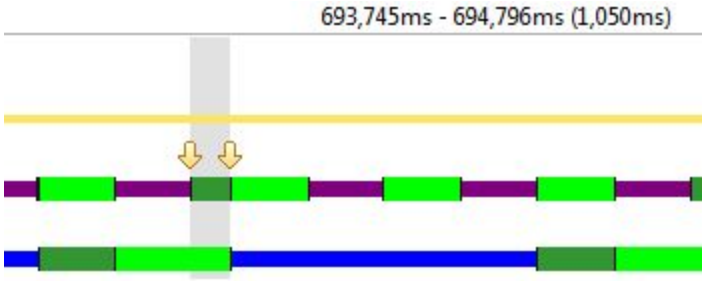


Jeweils nach 3 Takten (Iterationen) setzt Thread 1 ein Semaphore. Thread 2 wartet auf diesen. Bei einer theoretischen Zykluszeit von 4ms und 3 Iterationen, müsste Thread 1 alle 12 ms die Semaphore setzen und dadurch Thread 2 aufwecken. Der Versatz um 1ms ergibt sich aus dem single Prozessor System. Die 2te Iteration von Thread 1 kann erst mit einem Versatz von 1ms starten. Thread 2 ist in diesem Moment noch aktiv auf dem Prozessor.

Anmerkung:
Leider ist erst beim Verfassen des Dokuments der Fehler aufgefallen. Da kommenden Donnerstag 20.Juli Feiertag ist und unser regulärer Praktikumstermin (Mittwochs) die letzten beiden male nicht stattfindet, können wir keine neuen Screenshots anfertigen.

Fehler:
In unserer Implementierung berechnen wir den absoluten Zeitpunkt, bis zu dem der Thread nach der Verarbeitungszeit noch schlafen soll, am Anfang der Methode. Aktuelle Zeit + 4ms. Da Thread 1 aber erst mit einem Versatz von 1ms zum Zuge kommt, werden die 4ms auf den Zeitpunkt gezählt, an dem er an der Reihe ist. Dies führt dazu, dass manche Zyklen länger als die vorgegebenen 4ms dauern - siehe Abbildung, Abweichung um eine 1ms.

Lösung:
Den neuen Zeitpunkt an dem der Thread geweckt werden soll, bereits am Ende der vorherigen Iteration berechnen. So würde die 1ms bereits in die Zykluszeit der 4ms fallen.
Beispielrechnung: 1ms Warten (da noch Thread 2 aktiv) 2ms Verarbeitung und nur noch 1ms schlafen, da dann der Absolutzeitpunkt erreicht wäre.



Zusammenfassung – Ganze Time

Ein Ausschnitt der gesamten Timeline zeigt schön, wie Thread 2 (untere Timeline) immer auf das setzen der Semaphore von Thread 1 wartet (jeweils der blaue Balken)