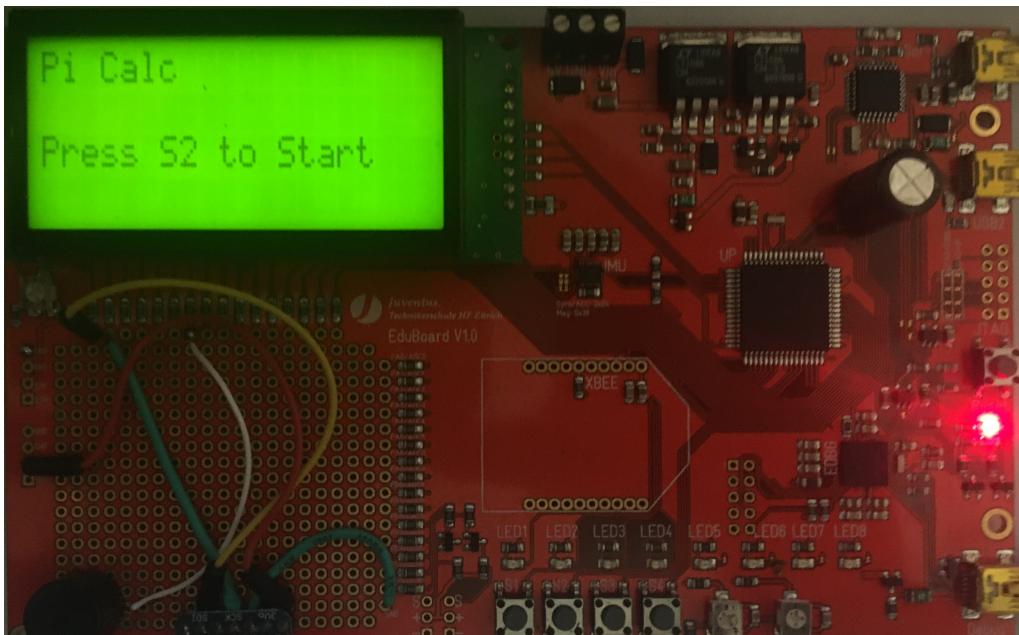


3.14159265358979323846264338327950288419716939937510582097·
0899862803482534211706798214808651328230664709384460955058·
1745028410270193852110555964462294895493038196442881097566·
8678316527120190914564856692346034861045432664821339360726·
0631558817488152092096282925409171536436789259036001133053·
5194151160943305727036575959195309218611738193261179310511·
673518857527249·
65664308602·
021798609437·
66940513200·
85771342757·
609173637·
87214684·
90122495343014654958537·
42019561·
2902196086·
344181598·
62977477130996051870721·
51059731·
281609631859·
24459455·
590830264252230825334468·
031378387528865875332·
381420617·
766914730359825349042875·
537875937519577818577·
532171224·
661300192787661119590921·
65485863278865936153·
182796823·
195203530185296899577362·
8347913151557485724·
6415069595·
953311686172785588907509·
550604009277016711·
0984882401·
836160356370766010471018·
374494482553797747·
471040475·
620804668425906949129331·
6205696602405803·
1935112533·
003558764024749647326391·
823547816360093·
6412199245·
03028618297455570674983·
90927210797509·
532116534493·
75596023648066549911988·
26542527862551·
1757467289097·
9380008164706001614524·
4144197356854·
6115735255213·
4184946843852332390739·
18983569485562692192218427255025·
25688767179049460165346·
8578438382796797668145410095388378636095068006422512520511·
269456042419652850221066118630674427862203919494504712371·
8746776465757396241389086583264599581339047802759009946576·
9570982582262052248940772671947826848260147699090264013639·

Pi Berechnungsprogramm - von Patrik Huber

FreeRTOS Basiert



Inhaltsverzeichnis:

Beschreibung	Seite 1
Umsetzung	Seite 1
Menüführung	Seite 2
Berechnung	Seite 6
Resultate	Seite 7

Beschreibung:

Bei diesem Projekt geht es um das Kennenlernen der Multitasking-Funktion des FreeRTOS-Betriebssystem auf dem ATxMega Mikroprozessor. Die Hauptaufgabe dieses Projektes war, die Zahl Pi auf 5 Stellen zu berechnen und die Zeit für die Berechnung misst.

Umsetzung:

Wie schon in der Beschreibung erwähnt, wird das FreeRTOS Betriebssystem eingesetzt und die Multitasking Funktion genutzt. Für die Menüführung wird ein Task aufgerufen, welcher für die Displayausgabe verantwortlich ist. Dieser Task schreibt die aktuellen Werte der Messung und Berechnung auf das Display. Wie das ganze visuell aussieht, wird in dem Kapitel „Menüführung“ gezeigt.

Die Tasterabfrage ist ein weiterer Task, welcher die Tasten abfragt und Befehle ausführt. Die Taster S1 und S2 werden für die Menüführung verwendet. Sobald Taster S1 gedrückt wird, springt das Menü ein Fenster zurück. Bei Taster S2 wird das nächste Fenster angezeigt. Hierfür wird eine globale Variabel verwendet.

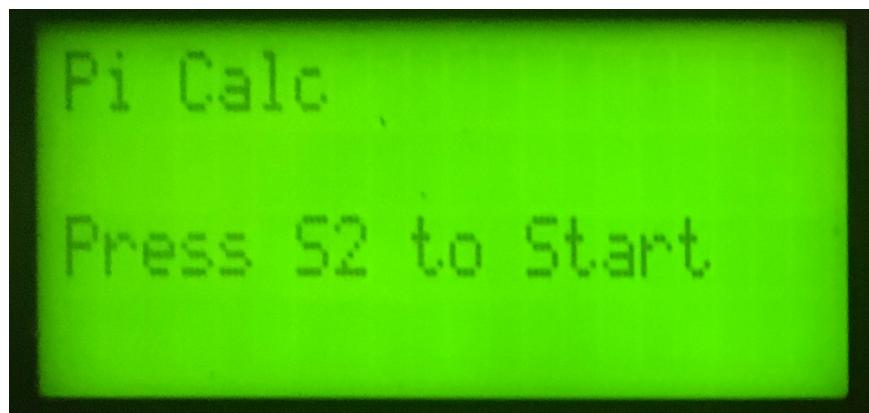
Die Taster S3 und S4 werden für die Start- und Stopbits verwendet. Diese setzen oder löschen Eventbits für andere Tasks. Somit kann der Timer genauer gestartet werden. Hierfür unten mehr.

Für die Zeitmessung wird ein Timer gewählt. Jedoch wird der TCC0 bereits vom Betriebssystem genutzt, und darum habe ich den Timer TCD0 gewählt. Dieser wird alle 1ms aufgerufen und zählt eine Variabel um 0.001 hoch. Diese Variabel hat einen Wert in der Einheit Sekunde und wird mittels dem Menütask auf dem Display angezeigt.

Um Pi zu berechnen wurden 2 Methoden angewendet.
Diese werden im Kapitel „Berechnung“ erklärt.

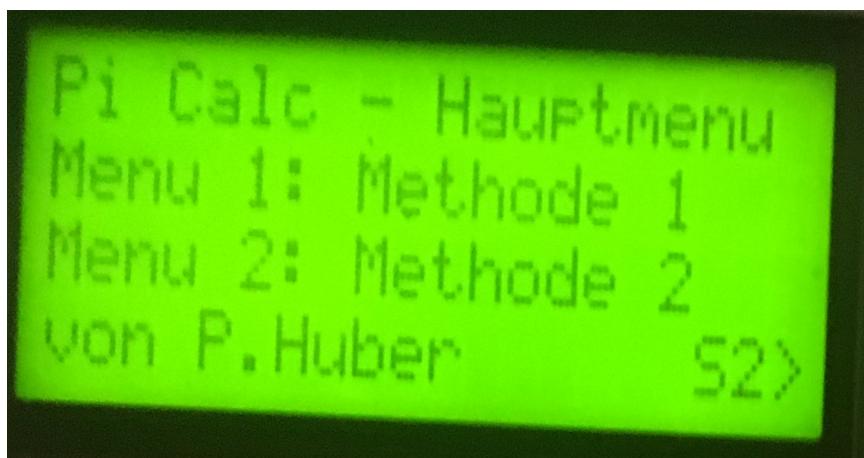
Menüführung

Sobald das Programm gestartet wird, wird ein Initialisierungs menü angezeigt.



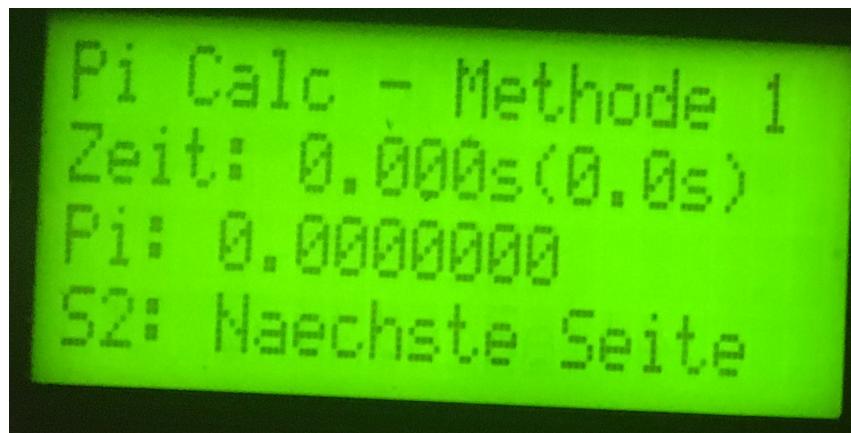
Durch drücken der Taste S2 wird das Programm gestartet. Der Timer ist jedoch noch ausgeschaltet und es wird noch nichts berechnet.

Sobald S2 gedrückt wurde, wird das Hauptmenü angezeigt. Hier wird eine Übersicht der Methoden angezeigt.



Durch drücken der Taste S2 wird das nächste Menü angezeigt.

Im Menü Pi Calc - Methode 1 wird die erste Berechnungsmethode angezeigt. Als erstes wird die Zeit angezeigt. Die Zahl ausserhalb der Klammer ist die gemessene Zeit, in der die Zahl Pi bis auf die 5. Kommastelle berechnet wurde. Die Zeit in der Klammer zeigt die aktuelle Zeit an.



Danach wird die Zahl Pi dargestellt bis auf die 7. Kommastelle genau.

Das Menü Pi Calc - Methode 2 ist gleich aufgebaut wie die erste Methode. Sie unterscheidet sich lediglich von der Berechnungsmethode, die dahinter steckt.



Wenn die letzte Methode aufgerufen werden soll, kann der Taster S1 gedrückt werden.

Berechnung

Für die Berechnung von Pi wurden 2 Methoden angewendet. Die erste Methode war, wie sich herausgestellt hatte, die langsamste Methode für die Berechnung. Daher habe ich mich entschieden 2 Methoden zu implementieren.

Methode 1:

Für die erste Methode habe ich mich für die Leibniz-Methode entschieden. Diese wird folgend berechnet:

$$\sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots = \frac{\pi}{4}$$

Dafür habe ich eine integer Variabel verwendet für die Irritation von k, welche sich nach jedem forever-Zyklus inkrementiert. Da die Potenzfunktion pow() von C lange dauert, habe ich mich dafür entschieden, nur das Vorzeichen zu wechseln. Jedoch gab das keinen grossen Unterschied. Am Schluss muss das Zwischenresultat mit 4 multipliziert werden und somit erhält man Pi.

Methode 2:

In der Methode 2 habe ich mich für die Methode von David H. Bailey, Peter Borwein und Simon Plouffe (1997) entworfen wurde entschieden. Diese habe ich nach längerem suchen im Internet gefunden.

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right)$$

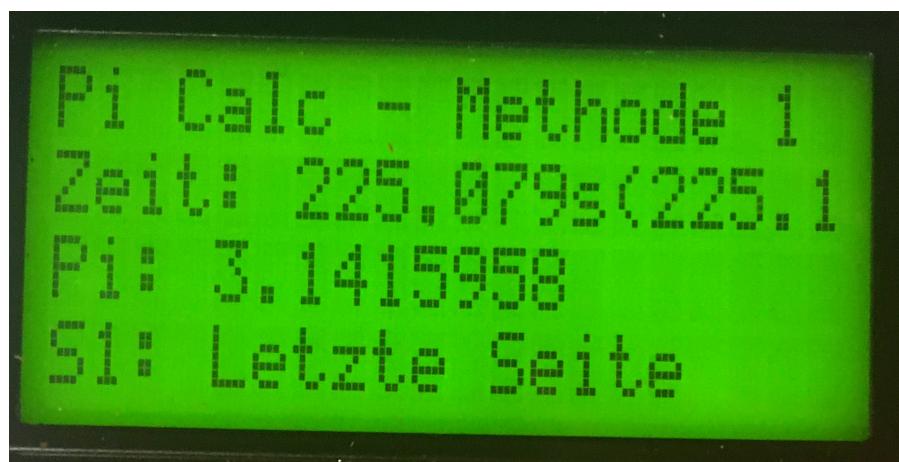
Auch bei dieser Methode wurde es gleich gehandhabt, wie die Methode 1. Der einzige unterschied war, dass das

Zwischenresultat nicht mehr mit einer Zahl multipliziert werden kann und kein Wechsel des Vorzeichen stattfindet. Somit gewann ich Geschwindigkeit.

Resultate

Als ich den Timer implementiert hatte, konnte ich die ersten Zeitmessungen der beiden Methoden durchführen.

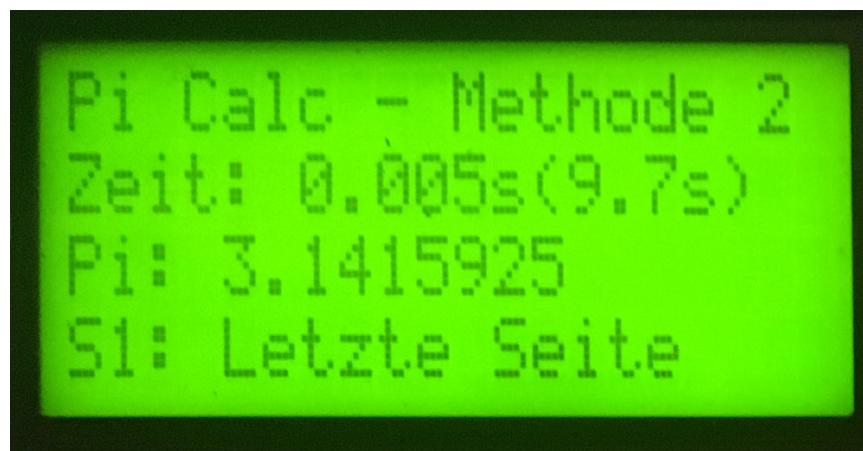
Methode 1:



Die erste Methode dauert 225.079 Sekunden, bis die berechnete Zahl bis auf die 5. Kommastelle mit Pi übereinstimmt.

Hier sieht man wie lange diese Leibnizmethode dauert. Man muss jedoch dazu sagen, dass die Verwendung der double-Variablen viele Rundungsfehler aufweist und dadurch die Zahl Pi nicht ganz genau berechnet werden kann.

Methode 2:



In der zweiten Methode, findet man schon sehr grosse Unterschiede gegenüber der Methode 1. Hier braucht die Berechnung von Pi, viel weniger Irritationen.