

Konzeptionsphase

PC-Anwendung:

Bei der PC-Anwendung habe mich hier für C++ entschieden, da hier viele Ressourcen und Bibliotheken zur Verfügung stehen und somit die Entwicklung vom Code einfacher ist. Ich habe mir vorgenommen das Programm wie folgt zu gestalten. Als erstes wird der Benutzer dazu aufgefordert die Verbindung zu dem Mikrokontroller über einem Button herzustellen. Nachdem die Verbindung erfolgreich passiert ist, kann der Benutzer die gewünschte Berechnung mit folgendem Format: $a+b$ | $a-b$ | $a*b$ | a/b eingeben. Kommazahlen können sowohl mit Punkt als auch mit Komma eingegeben werden. Nachdem die Eingabe passiert ist, werden die Ausdrücke separiert und nach Richtigkeit überprüft. Hier überprüft eine Funktion namens: „check_input()“ den eingegebenen String. Wenn die Funktion die Eingabe als ungültig bewertet, wird der Benutzer erneut dazu aufgefordert die Rechnung einzugeben. Nachdem der Benutzer eine korrekte Eingabe gemacht hat, wird der String an den Mikrocontroller gesendet und in einem Memofeld angezeigt. Es wird inzwischen die Verbindung zum μC , in einem Intervall von 100 Millisekunden, geprüft. Der Status der Kommunikation wird mit einer Anzeige im Layout kenntlich gemacht. Anschließend wird die Antwort bzw. das Ergebnis, welches vom Mikrocontroller zurückkommt, in einem Memofeld angezeigt. Es gibt ein Button, mit dem man die ganze Interaktion zwischen der PC - und μC -Anwendung speichern kann, hier wird einfach der ganze Memofeld-Inhalt in eine Textdatei übertragen. Zusätzlich gibt es einen Button, mit dem man die komplette Anwendung beenden kann.

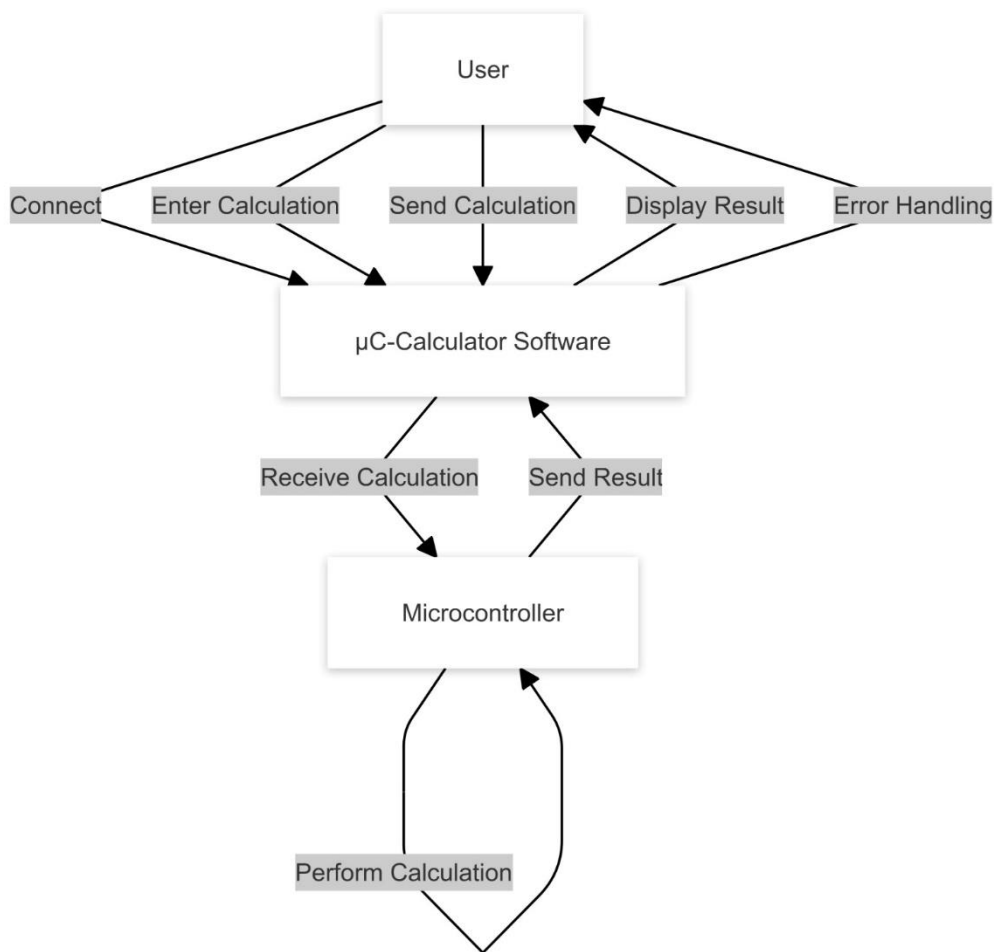
check_input() – Funktion:

- Diese Funktion überprüft, ob die Zeichenkette zwei Zahlen und einen gültigen Operator beinhaltet.
 - Erlaubte Operatoren: $+$, $-$, $*$, $/$
 - Es sind sowohl positive als auch negative Zahlen erlaubt. Zusätzlich wird überprüft, ob durch null geteilt wird.
 - Bei einer Kommazahl wird überprüft, wo sich der Punkt bzw. das Komma befindet.

Mikrocontroller-Anwendung:

Als Erstes werden die benötigten Variablen initialisiert und die Verbindung zum PC konfiguriert. Als Nächstes wird die Funktion „void loop()“ ausgeführt, hier wird gewartet bis eine Eingabe erfolgt. Nachdem eine Eingabe erkannt worden ist, wird die Zeichenkette in 3 einzelne Variablen namens: „Zahl1“, „Operator“ und „Zahl2“ separiert und entsprechend in die passenden Datentypen umgewandelt. Ist eine Kommazahl eingegeben worden, so wird das Komma in Punkt umgewandelt damit der Wert der Variable korrekt zugewiesen werden kann. Abhängig vom Operator wird die entsprechende mathematische Operation ausgeführt. Eine arithmetische Division durch null wird hier ebenso abgefragt. Auch wenn die Eingabe bereits bei der PC-Anwendung überprüft wird, habe ich mich dazu entschieden hier ebenso die Überprüfung einzubauen. Der Grund hierfür war die Überlegung, dass die μC -Anwendung für andere PC-Anwendungen, wo die Überprüfung möglicherweise nicht bereits erfolgt ist, ebenso funktional sein soll. Nachdem das Ergebnis ermittelt worden ist, wird die Zahl in eine Zeichenkette umgewandelt und zurück an die PC-Anwendung übermittelt. Der μC wartet auf die nächste Eingabe...

Use-Case Diagramm:



Sequenzdiagramm:

