

6.9.A Extras/Präprozessor – Arbeitsblatt

Aufgabe 1: Makros ohne Parameter

Erstellen Sie Makros für die folgenden Zahlen:

- a) EPSILON_0 für $8,85 \cdot 10^{-12}$ (Elektrische Feldkonstante)
- b) MILLI für 0,001
- c) MIKRO für 10^{-6}
- d) NANO für 10^{-9}
- e) PICO für 10^{-12}

Aufgabe 2: Makros mit Parametern

Erstellen Sie für die folgenden Zwecke Makros:

- a) KREISFREQUENZ (f) ergibt $2 \cdot \pi \cdot f$
- b) PRODUKT (a, b, c) ergibt $a \cdot b \cdot c$
- c) MAX (a, b) ergibt das Maximum der beiden Zahlen
- d) BETRAG (x) ergibt $-x$, falls x negativ ist, sonst x
- e) SINUS (x) ergibt $\sin x$, wobei x nicht im Bogenmaß, sondern in Grad verwendet wird, so dass SINUS (90) annähernd den Wert 0.0 haben sollte

Aufgabe 3: Makros mit und ohne Parameter

Das folgende Programm ist an die Programmiersprache PASCAL angelehnt, die auch bestimmend für das Aussehen von SCL bzw. ST wurde:

```

1 PROGRAM (neu)
2 VAR
3     int x;
4 BEGIN
5     FOR (x,0,4,1)
6     DO
7         printf("%i\n", x);
8     END;
9 END

```

- a) Bauen Sie Makros, die es erlauben, diesen Text mit einem C-Compiler zu compilieren!
- b) Bauen Sie ein Makro, das es erlaubt, im Programm statt der printf-Anweisung die Anweisung writeInt(x) zu benutzen!

Aufgabe 4: Bedingte Compilierung

Erstellen Sie auch hier passende Präprozessor-Direktiven:

- a) M_PI – Kreiszahl π : Falls M_PI noch nicht als Makro definiert wurde (durch Einbindung von <math.h>), dann soll es als Makro für den Wert 3.141592654 definiert werden.
- b) M_E – Euler'sche Zahl e : Falls M_E noch nicht als Makro definiert wurde (durch Einbindung von <math.h>), dann soll es als Makro für den Wert 2.718281828 definiert werden.