

## 1. Befehlssatz [je 2 Pkt.]

- a) Registerinhalte: D0 = \$10204080      D1 = \$80406090      Flags: C:0 Z:0 N:0  
 b) Registerinhalte: D0 = \$10204080      D1 = \$80400000      Flags: C:0 Z:1 N:0

## 2. Adressierungsarten [je 2 Pkt.]

- a) MOVE.B 11(A0),D1      A0 = \$00200A88      D1 = \$AABBCC35  
 b) MOVE.L -(A0),D1      A0 = \$00200A84      D1 = \$16171819  
 c) MOVE.W -10(A0,D0.W),D1      A0 = \$00200A88      D1 = \$AABB1819  
 d) MOVE.W #\$12,D1      A0 = \$00200A88      D1 = \$AABB0012

## 3. Gegeben ist folgendes Programm: [8 Pkt.]

```

MOVE.L #Tab,A0      Zeiger auf eine Tabelle in Adressregister A0
MOVEQ.L #3,D3      Konstante 3 in Datenregister D3
CLR.L D2      Datenregister D2 löschen
Loop: MOVE.W (A0)+,D0      Element auf das der Zeiger A0 zeigt holen und in D0, Zeiger
                           auf nächsten Tabelleneintrag setzen (A0=A0+2)
                           MOVE.W (A0)+,D1      Element auf das der Zeiger A0 zeigt holen und in D0, Zeiger
                           auf nächsten Tabelleneintrag setzen (A0=A0+2)
                           MULU D0,D1      Multiplizieren der beiden Einträge
                           ADD.L D1,D2      Addieren zur Gesamtsumme
                           DBF D3,Loop      Schleife 4x durchführen (D3+1)
                           ... (Weiterer Programmablauf)
Tab: DC.W 1,20,2,10,1,8,3,20,5,8

```

- a) Was macht dieses Programm? Erklärung der einzelnen Schritte.[4]  
 Das Programm multipliziert ungerade mit geraden Einträgen einer Tabelle und addiert die Produkte.
- b) Wieviele Male läuft die Schleife durch? [1]      Schleife läuft 4x durch
- c) Welchen Inhalt nimmt D2 an (dezimal oder hexadezimal), wenn die Schleife beendet ist? [3]      108d

## 4. Programmieraufgabe [8 Pkt.]

Schreiben sie ein Programm, das in der Anzeige die Anzahl gedrückter Tasten in der oberen Tastenreihe in der rechten Anzeige anzeigt.

```

*Zählt die Anzahl der gedrückten Tasten      DBF D2,Loop
OPT 1      MOVE.B D3,Disp2 *Auf Anzeige
ORG $200800      * [1 Punkt]      JMP Start *Back to start

Disp1 EQU $200300
Disp2 EQU $200200
LED EQU $200100
Tast EQU $200100

Start: MOVE.B Tast,D0 *Tasten einlesen
        MOVE.B D0,D1 *Kopie anfertigen
        NOT.B D1 *Invertieren
        MOVE.B D1,LED *Ausgabe auf LED
        CLR.B D3 *D3 = Zähler = 0
        MOVE.W #7,D2 *7 Bit
Loop: MOVE.B D0,D1 *Kopie machen
        AND.B #%00000001,D1 *Nur 1 Bit
        ADD.B D1,D3 *Zum Zähler
        LSR.B #1,D0

```