# Final Exam

# Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Time limit is 120 minutes.
* Exam is open book and notes and calculators are permitted.
* Write your answers in the answer blanks provided.

1. 24 points – Addressing modes - For each of the following assembly instructions, identify the addressing mode used as EXT, DIR, IMM, INH, IDX, IDX1, IDX2, REL. Assume the labels MAX and Array are defined as follows:

MAX EQU 100

ORG $2000

Array DS.B 50

1. LDAB MAX Addressing mode:
2. LDX #Array Addressing mode:
3. CMPB Array Addressing mode:
4. STAB 25, X Addressing mode:
5. PSHX Addressing mode:
6. BRA done Addressing mode:

1. 20 points – Comparison Branching - For each of the code fragments below, determine which of the five given branches are taken (T), not taken (NT), or cannot be determined (CBD). Note the Bxx is used to represent a generic branch instruction.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | BHI | BLO | BGT | BLT | BEQ |
| LDAA #$D0  CMPA #$70  Bxx 10 |  |  |  |  |  |
| LDAA #$99  NEGA  CMPA #$80  Bxx 10 |  |  |  |  |  |
| LDAA #$90  ADDA #$FF  CMPA #$80  Bxx 10 |  |  |  |  |  |
| LDAA #$B0  LSLA  ASRA  CMPA #$30  Bxx 10 |  |  |  |  |  |

1. 20 points – Assembly to machine code translation – Convert the following assembly program into machine code using the most efficient addressing mode available (i.e. if possible, use direct instead of extended).

Indexed Addressing Postbyte Information

|  |  |
| --- | --- |
| Register | rr |
| X | 00 |
| Y | 01 |
| SP | 10 |
| PC | 11 |

Postbyte for 5-bit Offset: rr0nnnnn

Postbytes for 9-bit Offset: 111rr00n nnnnnnnn

Postbytes for 16-bit Offset: 111rr010 nnnnnnnn nnnnnnnn

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | Addresses | Contents |
|  | **ORG** | $C000 |  |  |  |
| sqrtloop: | CPD | 2,SP |  |  |  |
|  | BHI | donesqrt |  |  |  |
|  | ADDD | 0,SP |  |  |  |
|  | INC | 1,SP |  |  |  |
|  | ADDD | 0,SP |  |  |  |
|  | BRA | sqrtloop |  |  |  |
| donesqrt: | LDD | 0,SP |  |  |  |
|  | DECB |  |  |  |  |
|  | LEAS | 4,SP |  |  |  |
|  | RTS |  |  |  |  |
|  | | | | | |
|  |  |  |  |  |  |

1. 20 points – C and Assembly – Answer each question below.
   1. Why do we need to state the type of variables (as unsigned or signed) when declaring variables in C but not in Assembly?
   2. Write the Assembly language equivalents of the following C declarations? Recall that an int is 16 bits.

#define MAX 100

signed int sum

unsigned char grades[50]

* 1. Write a C ***function prototype*** for a function called “add2ints” that adds two ints N1 and N2 both passed by value, stores the result Answer as an int passed by reference, and returns by value a one-byte unsigned char that gives an error code. **Implementation of the function is not needed.**

1. 16 points – Assume that the following statements have already been defined for all problems.

PORTB EQU $0001

DDRB EQU $0003

PTP EQU $0258

DDRP RQU $025A

PTH EQU $0260

DDRH EQU $0262

PPSH EQU $0265

PIEH EQU $0266

PIFH EQU $0267

1. Perform the following operations using assembly without affecting any other bits.
   1. Configure pins 6, 5, 4, and 3 of Port H to outputs and the others to inputs
   2. Configure all pins of Port B to inputs
   3. Configure pins 1 and 2 of Port B to inputs
   4. Configure pins 2, 3, and 4 of Port P to inputs