Lösningsförslag

Uppgift 1:
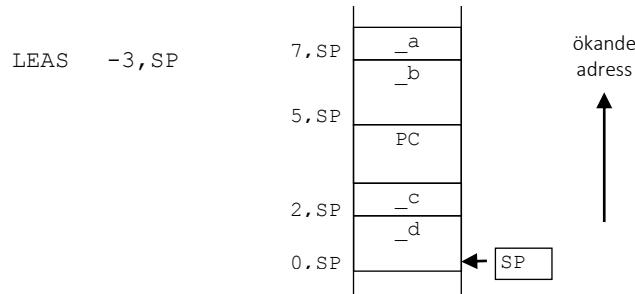
a)
```
_a      RMB    2
_b      RMB    1
```

b)
```
        LDAB   _b
        PSHB
        LDD    _a
        PSHD
        JSR    _func
        LEAS   3,SP
```

c)
```
        LEAS   -3,SP
```


d)

```
;   c = a;
        LDAB   7,SP
        STAB   2,SP
;   d = b;
        LDD    5,SP
        STD    0,SP
;       c = *b;
        LDAB   [5,SP]
        STAB   2,SP
;   d = &a;
        LEAX   7,SP
        STX    0,SP
```

Uppgift 2:

```
    EXPORT _install_irq_handler    [r,2]
;    IrqHandler * install_irq_handler( IrqHandler irq, IrqHandler ** vector)
_install_irq_handler:
;    {
    LEAS   -2,SP
;      IrqHandler * p;
;      p = *vector;
    LDD    [6,SP]
    STD    0,SP
;       *( (IrqHandler *) vector) = irq;
    LDD    4,SP
    STD    [6,SP]

    CLI

;      return p;
    LDD    0,SP
;    }
    LEAS   2,SP
    RTS
```

Uppgift 3:

```
#define ML4OUTH     *(( unsigned char *) 0x400)
#define ML4OUTL     *(( unsigned char *) 0x401)
#define ML4IN       *(( unsigned char *) 0x600)
```

```
void  DisplayNBCD( void )
{
   char c;
   c = ML4IN;
   if( c < 10 )
   {
      c = c*c;  /* kvadrera */
      ML4OUTH = SegCodes[c / 10 ];  /* mest signifikant */
      ML4OUTL = SegCodes[c % 10 ];  /* minst signifikant */
   }else{
      ML4OUTH = ERROR_CODE;  /* felkod till båda indikatorer */
      ML4OUTL = ERROR_CODE;
   }
}
```

Uppgift 4:

```
; Adressdefinitioner
SWITCH1   EQU    $600
SWITCH2   EQU    $601
DISPLAY   EQU    $400

; Subrutin
AddUnsigned8bitTo16:
          LDAB   SWITCH1
          CLRA
          PSHD
          LDAB   SWITCH2
          ADDD   2,SP+
          STD    DISPLAY
          RTS
```

Uppgift 5:

```
char *strcat(char *s1, const char *s2)
{
   char *save = s1;

   while (*s1 != 0)
      s1++;
   while (*s2 != 0)
      *s1++ = *s2++;
   *s1 = 0;
   return(save);
}
```

Uppgift 6:

```
// Filen ports.h
#define ML4OUT_ADR 0x400
#define ML4OUT     *((unsigned char *) ML4OUT_ADR)

// Filen stegmotor.h
void motor_init();
void motor_vrid(int antal_steg, int medurs);

// Filen stegmotor.c
#include "stegmotor.h"
#include "ports.h"
#include "clock.h"     // innehåller deklaration av funktionen hold

static unsigned char tillstand[] = {0x90, 0xA0, 0x60, 0x50};
static int aktuellt_tillstand= 0;

void motor_init()
{
        aktuellt_tillstand = 0;
}
```

```
void motor_vrid(int antal_steg, int medurs)
{
        int steg = (medurs) ? 1 : -1;
        int i;
        for (i=0; i<=antal_steg; i++) {
                aktuellt_tillstand =(aktuellt_tillstand+4+steg) % 4;
                ML4OUT = tillstand[aktuellt_tillstand];
                hold(10);
        }
}
```