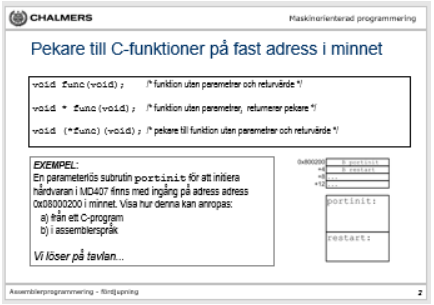
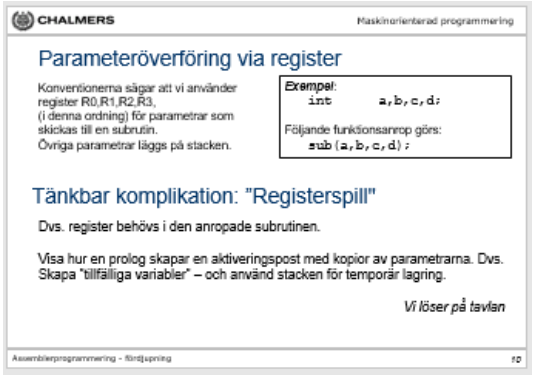


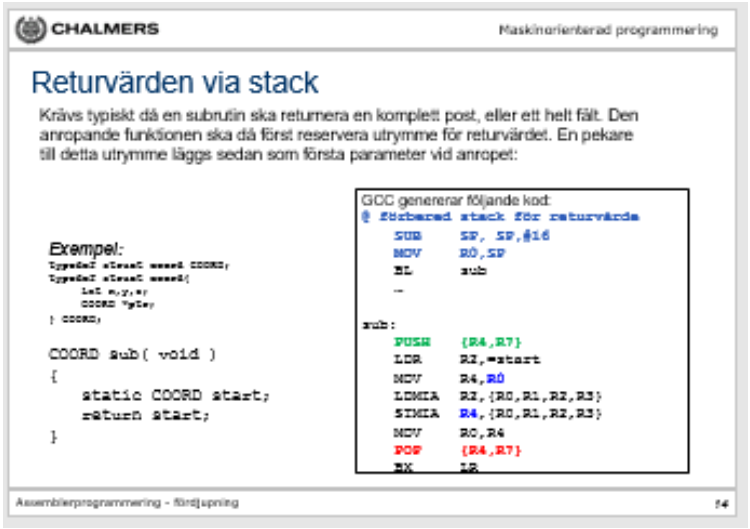

```
a)
typedef     void (*simplefunc)(void);
/* obs Thumb-bit i adressbit 0 */
#define     portinit ((simplefunc) 0x08000201 )
void xxx( void )
{
      ...
      portinit();
      ...
}
b)
      ...
      LDR   R0,=0x8000201 @ obs Thumb-bit i A0...
      BLX   R0
```




```
Alt 1: SP är "framepointer"
sub:
@ parametrar finns i register,
@ spara dessa på stacken
        PUSH    {LR}
        SUB     SP,SP,#20
        STR     R0,[SP,#12]
        STR     R1,[SP,#8]
        STR     R2,[SP,#4]
        STR     R3,[SP]
        ...
        ADD     SP,SP,#20
        POP     {PC}
```

```
Alt 2: R7 är "framepointer"
sub:
@ parametrar finns i register,
@ spara dessa på stacken
        PUSH    {LR,R7}
        SUB     SP,SP,#16
        MOV     R7,SP
        STR     R0,[R7,#12]
        STR     R1,[R7,#8]
        STR     R2,[R7,#4]
        STR     R3,[R7]
        ...
        MOV     SP,R7
        ADD     SP,SP,#16
        POP     {PC,LR}
```

Förklaring till instruktionerna LDMIA och STMIA




```
LDMIA: Load Multiple Increment After
STMIA: Store Multiple Inclement After

R2 = start
LDMI              R2,{R0,R1,R2,R3}
->
         (start)  ->R0
         (start +4)->R1
         (start +8)->R2
         (start +12)->R3
         R2 <- start + 16

R4 = dest
STMIA    R4,{R0,R1,R2,R3}
->
         (dest)            <- R0
         (dest +4)         <- R1
         (dest +8)         <- R2
         (dest +12)        <- R3
         R4 <- dest + 16
```