Lecture 15: Subroutines
Today’s Topics

• What is subroutines?

• Learn how to call subroutines from an assembly program.

• Learn the properties of well-written subroutines.

• Learn how to use pass-by-reference and pass-by-value to send parameters to a subroutine.
What is Subroutines?

A definition of subroutines

• A subroutine in a self-contained section of code that implements a specific function that can be called from many different places.

• Reasons why we use subroutines
  ▪ Save memory
    • The amount of memory required to store a program is reduced if the code to implement a function is stored only once instead of each time the function is needed in the program.
  ▪ Improve reusability of code
    • Functions implemented subroutines are often easier to insert into future programs for code reuse.
  ▪ Better organization
    • A complicated program can be more organized with subroutines.
    • Execution time increases and code size may increase as well.
Need Something for Subroutine Operation

Let’s take a look at this code.

```
ORG $2000
LDAA #17 ; 2000
BRA MagA ; 2002

Ret1 STAA $1000 ; 2004
LDAA #-1 ; 2007
BRA MagA ; 2009

Ret2 STAA $1001 ; 200B
SWI ; 200E

; Subroutine MagA
; compute magnitude of a single byte number
; input: byte in register A
; output: magnitude returned in register A

ORG $2200
MagA TSTA
BPL return
NEGA
Return BRA ??????
```
Subroutine Instructions

JSR and RTS

• JSR (Jump Sub Routine)
  ▪ Pushes two-byte address of the next line of code on the stack first.
  ▪ Jump/Branch to the subroutine.

• RTS (ReTurn from Subroutine)
  ▪ Pulls two bytes off the stack and jumps to that address.
Example for Subroutines

```
ORG $2000
LDS #$3600  ; 2000
LDAA #17   ; 2003
                
Jsr1  JSR MagA   ; 2005
Ret1  STAA $1000; 2008
                
                
Jsr2  JSR MagA   ; 200D
Ret2  STAA $1001; 2010
                
SWI       ; 2013
                
; compute magnitude of a single byte number
; input: byte in register A
; output: magnitude returned in register A
MagA TSTA
                
BPL  return
NEGA
Return RTS
                
35FD XX 35FD XX 35FD XX 35FD XX
35FE 20 35FE XX 35FE 20 35FE XX
35FF 08 35FF XX 35FF 10 35FF XX
3600 XX 3600 XX 3600 XX 3600 XX

SP 35FE SP 3600 SP 35FE SP 3600
```
Nesting Subroutines

ORG $2000
LDS #$3600 ; 2000
LDAA #17 ; 2003
LDAB #-1 ; 2005
JsrAB JSR MagAB ; 2007
SWI ; 200A

MagAB JSR MagA ; 200B
PSHA1 PSHA ; 200E
TFR B,A ; 200F
JSR1 JSR MagA ; 2011
TFR A,B ; 2014
PULA1 PULA ; 2016
RTS1 RTS ; 2017

MagA TSTA ; 2018
BPL return ; 2019
NEGA ; 201B
Return RTS ; 201C

<table>
<thead>
<tr>
<th>After</th>
<th>After</th>
<th>After</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td>JsrAB</td>
<td>MagAB</td>
<td>Return</td>
<td>PSHA1</td>
</tr>
<tr>
<td>35FB</td>
<td>XX</td>
<td>35FB</td>
<td>XX</td>
</tr>
<tr>
<td>35FC</td>
<td>XX</td>
<td>35FC</td>
<td>XX</td>
</tr>
<tr>
<td>35FD</td>
<td>XX</td>
<td>35FD</td>
<td>XX</td>
</tr>
<tr>
<td>35FE</td>
<td>20</td>
<td>35FE</td>
<td>20</td>
</tr>
<tr>
<td>35FF</td>
<td>0A</td>
<td>35FF</td>
<td>0A</td>
</tr>
<tr>
<td>3600</td>
<td>XX</td>
<td>3600</td>
<td>XX</td>
</tr>
</tbody>
</table>

SP 35FE  SP 35FC  SP 35FE  SP 35FD
Nesting Subroutines –cont’d

<table>
<thead>
<tr>
<th>After Jsr1</th>
<th>After Return</th>
<th>After PULA1</th>
<th>After RTS1</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>35FB</td>
<td>20</td>
<td>35FB XX</td>
<td>35FB XX</td>
<td>35FB XX</td>
</tr>
<tr>
<td>35FC</td>
<td>14</td>
<td>35FC XX</td>
<td>35FC XX</td>
<td>35FC XX</td>
</tr>
<tr>
<td>35FD</td>
<td>11</td>
<td>35FD 11</td>
<td>35FD XX</td>
<td>35FD XX</td>
</tr>
<tr>
<td>35FE</td>
<td>20</td>
<td>35FE 20</td>
<td>35FE 20</td>
<td>35FE XX</td>
</tr>
<tr>
<td>35FF</td>
<td>0A</td>
<td>35FF 0A</td>
<td>35FF 0A</td>
<td>35FF XX</td>
</tr>
<tr>
<td>3600</td>
<td>XX</td>
<td>3600 XX</td>
<td>3600 XX</td>
<td>3600 XX</td>
</tr>
</tbody>
</table>

SP 35FB  SP 35FD  SP 35FE  SP 3600
Well-written Subroutines**

- One entry point
- One exit point
- One specific function
- One returned object
- Use the stack to store local variables
Parameter Passing

• Pass-by-value
  ▪ The data itself is passed

• Pass-by-reference
  ▪ The address of the data is passed
Questions?
Wrap-up

What we’ve learned

• Subroutines

• JSR, RTS
What to Come

- Parameter passing