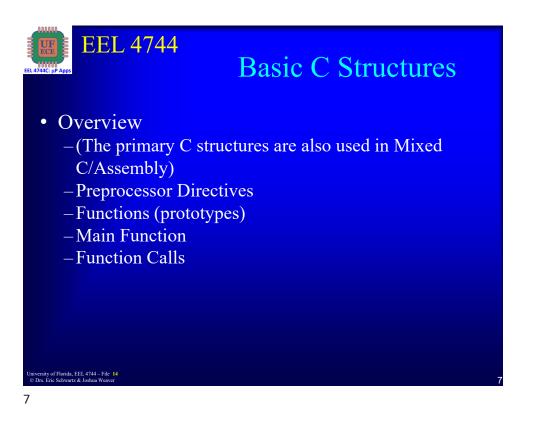


EEL 4744 What is Mixed C/Assembly Coding? Use Assembly code to improve C code or take advantage of a specific processor's capabilities For our board, mixed coding is handled by the AVR-GCC toolchain for compiling with the GNU Assembler (GAS); previously used Eclipse toolchair. W/ XMEGA, 2 ways to mix C and Assembly code Use separate files for C code and Assembly code, the .c extension and .s extension respectively Inline Assembly code directly inserted into the C code

> You will **NOT** be expected to write mixed code

See the end of this lecture for more mixed C/Assembly info
 niversity of Forida, EEL 1744-File 14
 Dis. Ent Schwarz & Johan Weaver

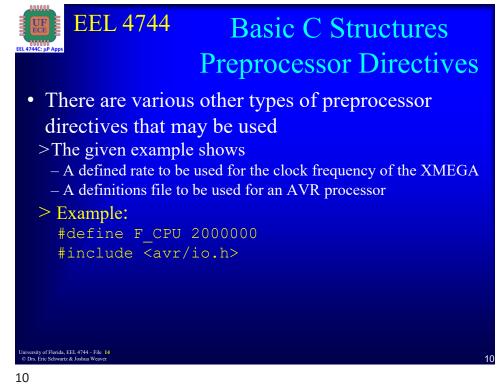
6



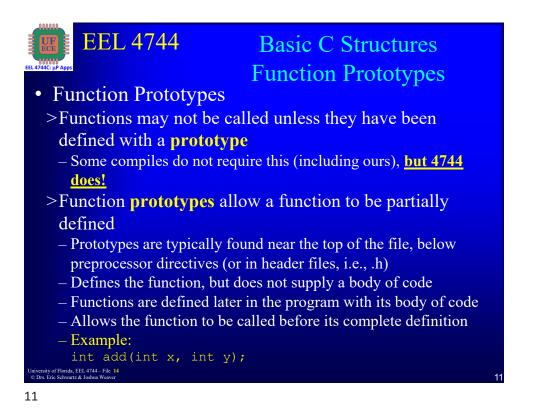
EEL 4744 Ba	sic C Structures			
• C (or Mixed C/Assembly)				
start with a standard	// main routine below			
structure	int main(void) {			
> Example:	int x=3, y=7, z;			
	while(1)			
#define F_CPU 2000000	{			
<pre>#include <avr io.h=""></avr></pre>	z=add(x, y);			
	}			
// function prototype below	}			
<pre>int add(int x, int y);</pre>	// function is below			
	<pre>int add(int x, int y)</pre>			
	{			
	return (x+y);			
University of Florida, EEL 4744 – File 14 © Drs. Eric Schwartz & Joshna Weaver	3			

8



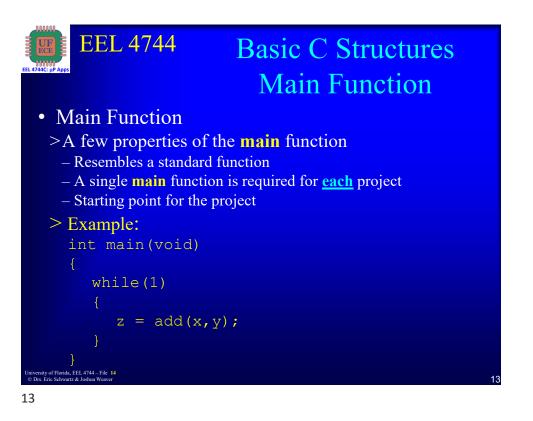


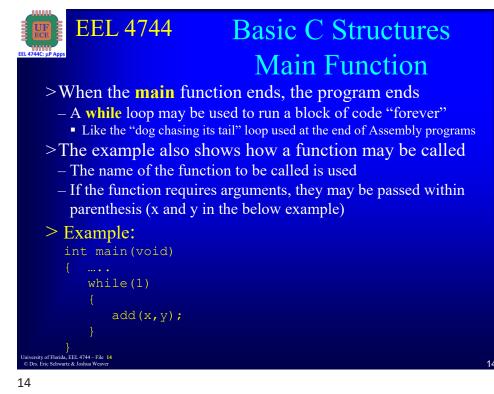
5

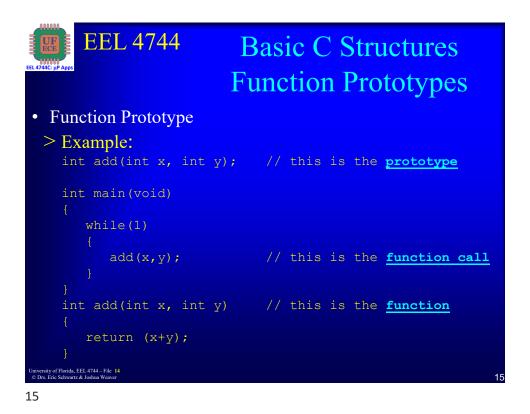


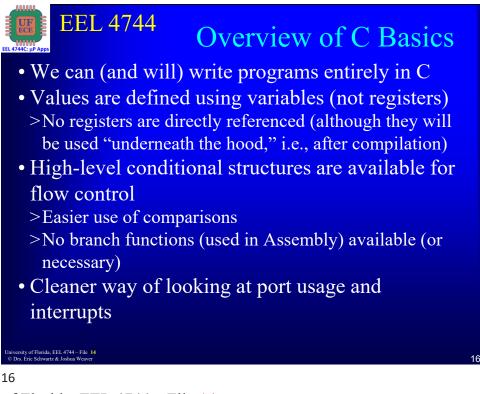


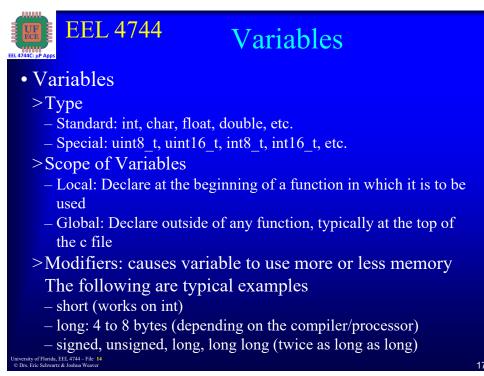
© Drs. Eric Schwartz & Joshua Weaver

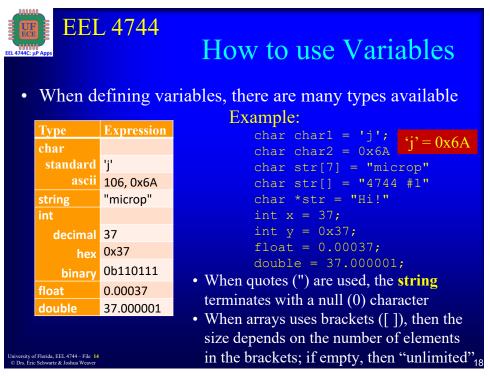




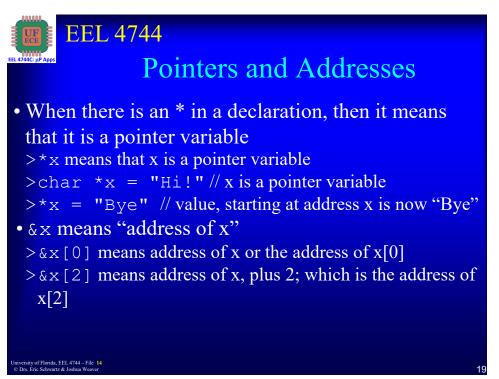


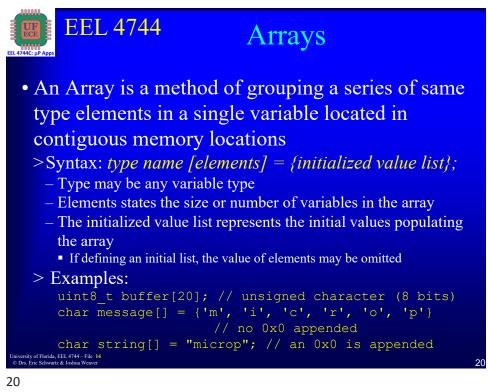


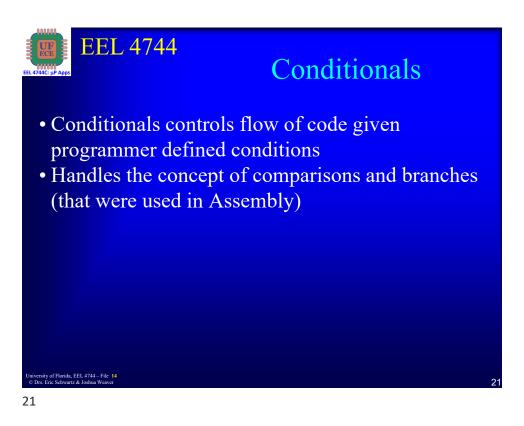


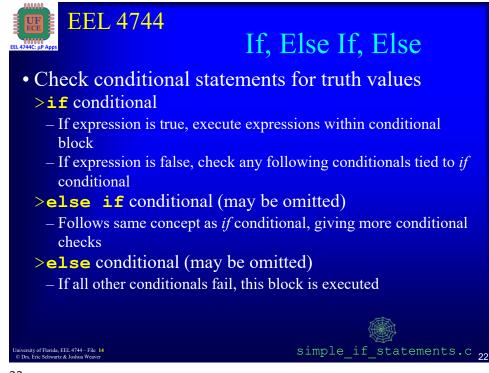








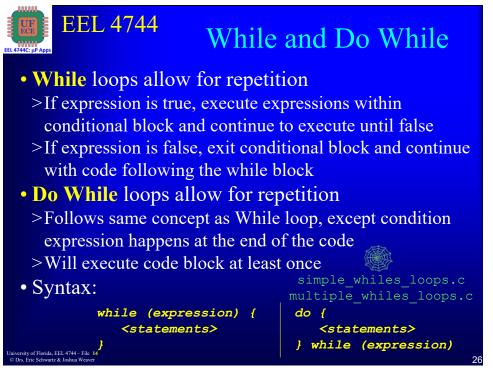


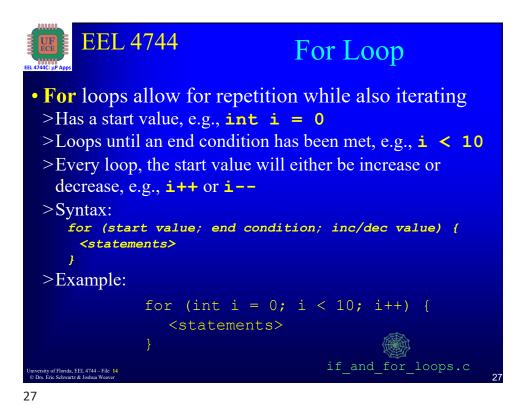


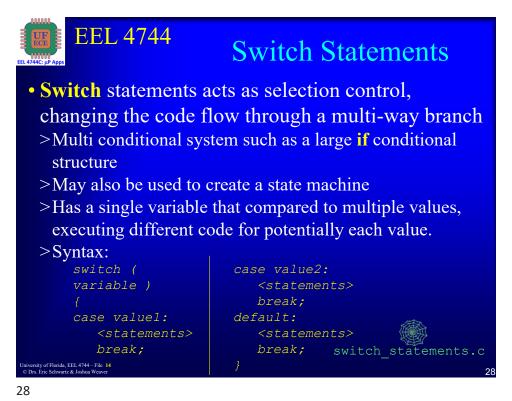


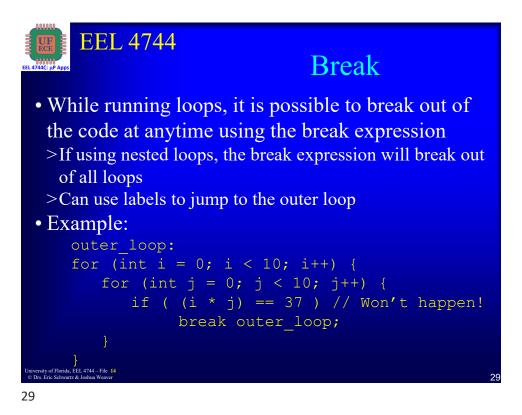
EEL 4744 Relational Operators • To create a conditional expression, utilize one of relational operators				
	Relational Operator	Definition	Example (True results)	
	>	Greater than	47 > 37	
	>=	Greater than or equal to	47 >= 47	
	<	Less than	37 < 47	
	<=	Less than or equal to	37 <= 47	
	==	Equal to	47 == 47	
	! =	Not equal to	37 != 47	
iiversity of Florida, EEL 4744 – File 14 Drs. Eric Schwarz & Johna Weaver 22				

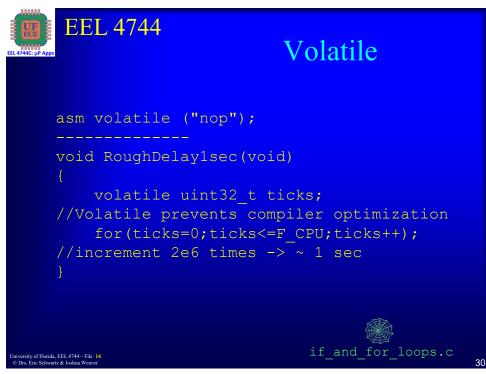
EEL 4744 Boolean Operators • To create more complex conditional expressions, Boolean operators may be used			
	Boolean Operator	Definition	Example (True results)
	& &	AND two expressions	((47 >= 47) && (47 > 37))
		OR two expressions	((37!=47) (37>47))
	!	Complement expression	!(37>47)
	y of Florida, EEL 4744 – File 14 Eric Schwartz & Joshua Weaver		24

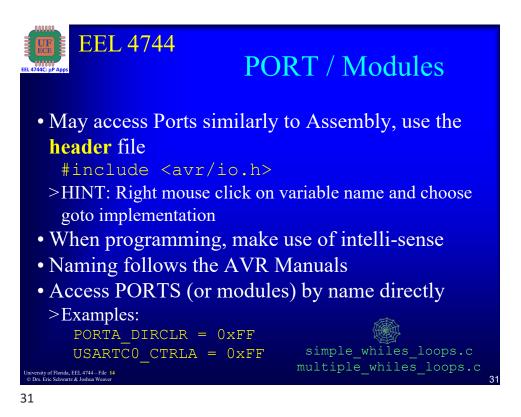




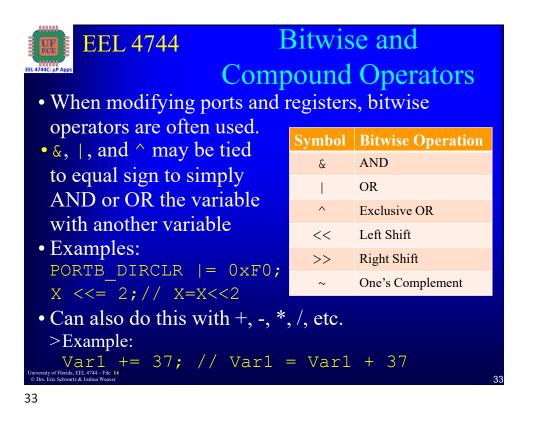


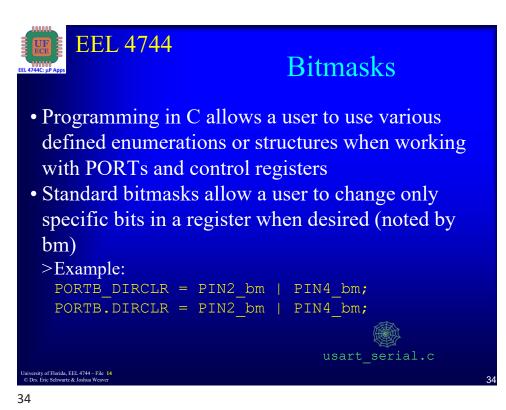




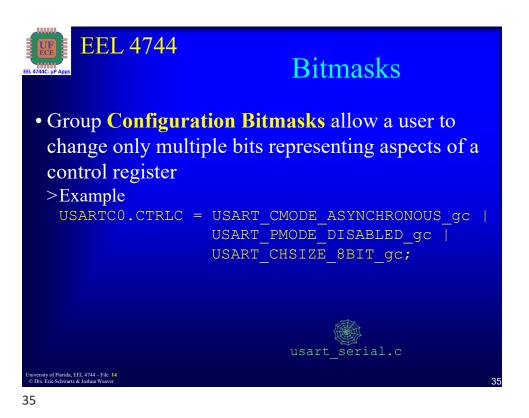


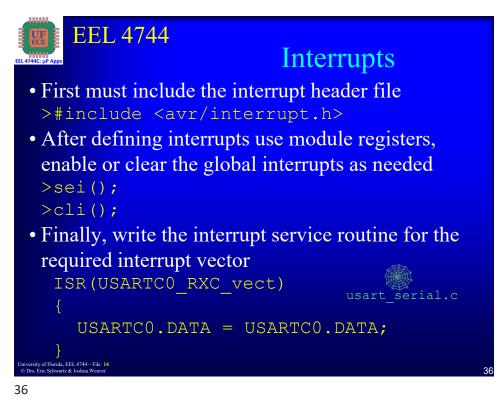


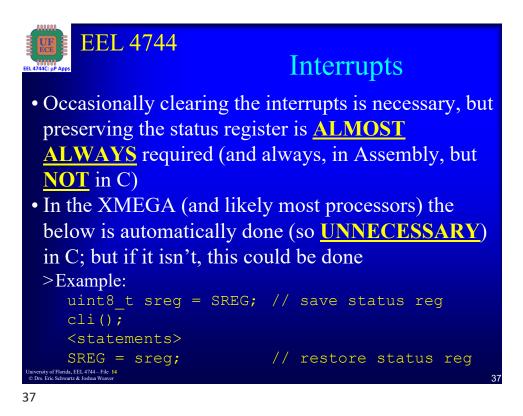


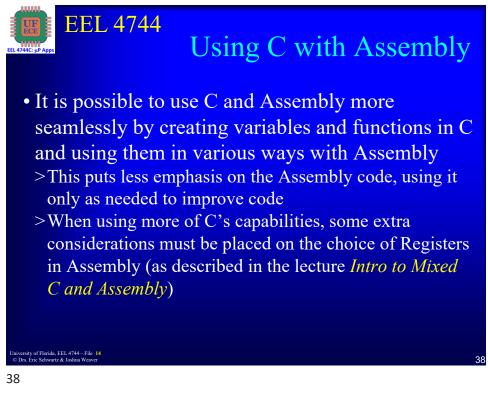


Intro to C for XMEGA

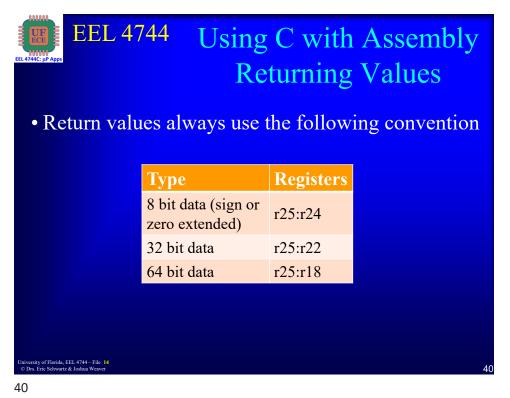


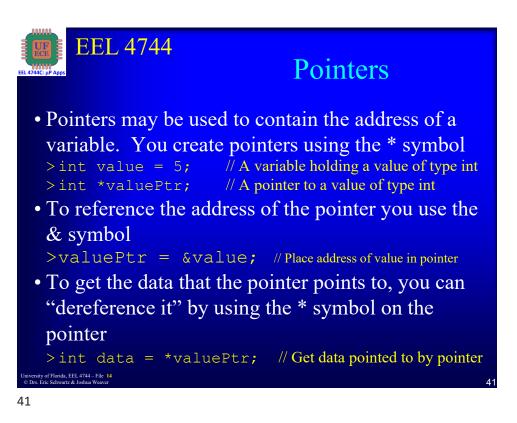


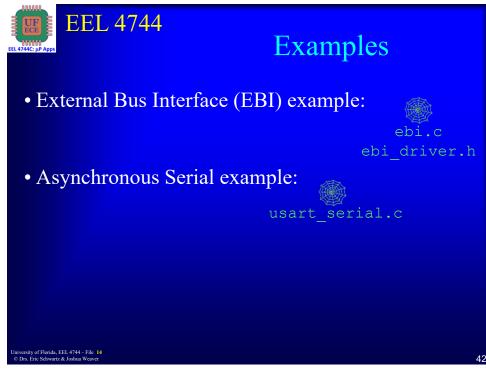




 EEL 4744 Using C with Assembly Dassing Arguments Arguments are passed to Assembly functions in register pairs or via the stack if more than 9 arguments Word Data takes both registers Byte Data takes the lower register 			
	Argument	Registers	
	1	r25:r24	
	2	r23:r22	
	3	r21:r20	
Thinkin (Phoise FFT 474) FFt 44	9	r9:r8	
University of Florida, EEL 4744 – File 14 © Drs. Eric Schwartz & Joshua Weaver			39



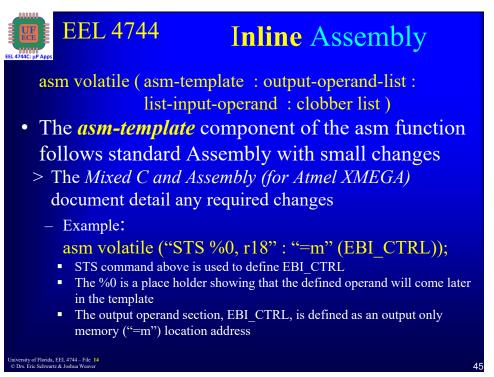


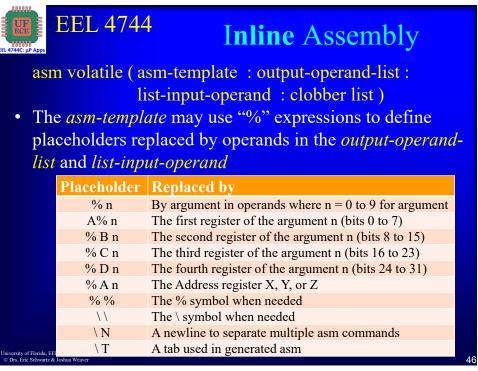




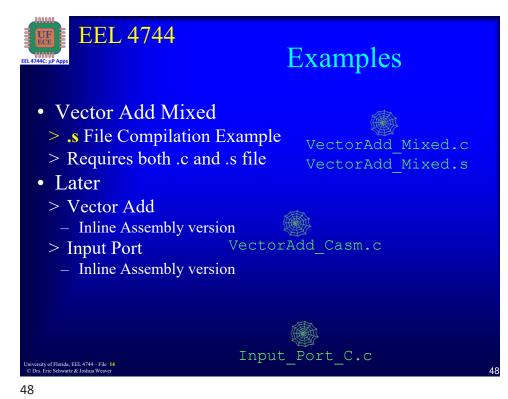


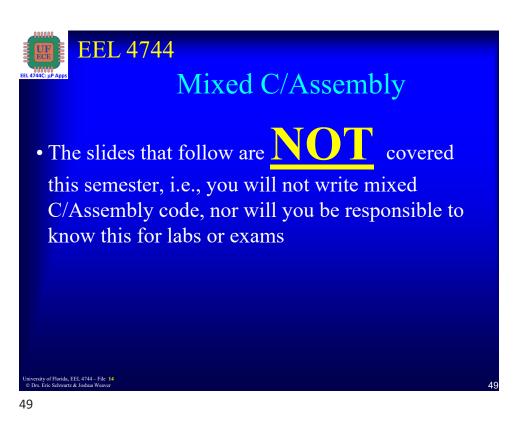
EEL 4744 C Projects and Inline Assembly • If it is only desired to add a few lines of assembly code to a C Project, it is possible to add assembly "inline" • Inline assembly uses the asm function with the following template asm volatile(*asm-template* : *output-operand-list* : *list-input-operand* : *clobber list*) • When using the **asm** function, the compiler will have a harder time optimizing code • The volatile keyword may be used to prevent the compiler from attempting to optimize the line > The keyword volatile may be omitted, but then the compiler might optimize away your intended structure 44

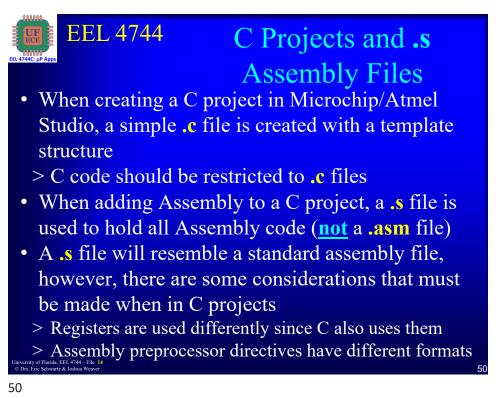




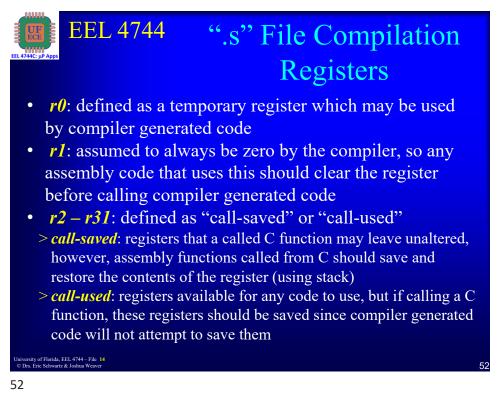
EEL 4744C: IN Apps	. 4744	Inline Assen	nbly	
 asm volatile (asm-template : output-operand-list : list-input-operand : clobber list) The <i>output-operand-list</i> and <i>list-input-operand</i> uses various modifiers as needed for the operands given 				
	Modifier	Meaning		
	=	Output operand		
	&	Not used as input but only an output		
	+	Input and Output Operand		
University of Florida, EEL 4744 File 14 © Drs. Eric Schwartz & Joshua Weaver			47	
0 Drs. Eric Schwartz & Joshua Weaver			47	



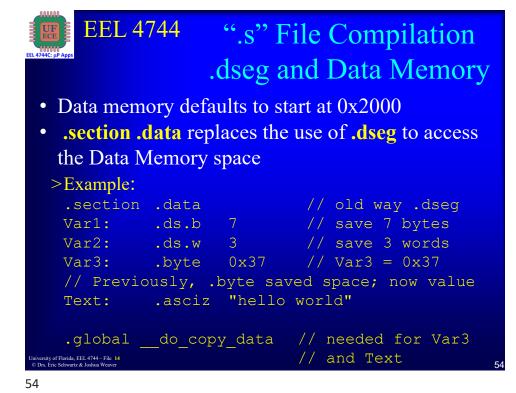


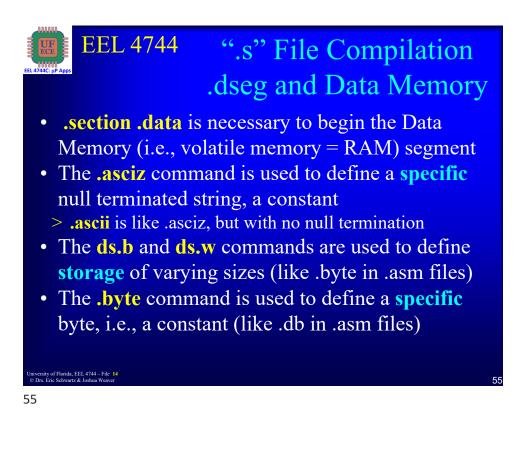


EEL 4744 ".s" File Compilation Registers When writing assembly in a C project, registers have			
Register	Description	Assembly code called from C	Assembly code that calls C code
r0	Temporary	Save and restore	Save and restore
r1	Always Zero	Must clear before returning	Must clear before returning
r2-r17 r28 r29	"call-saved"	Save and restore	Can freely use
r18-r27 r30	"call-used"	Can freely use	Save and restore



• When writing assembly in a C project, some EEL 4744 ".s" File Compilation Syntax Path for the equivalent file to the ATxmega128A1Udef.inc is C:\Program Files (x86)\Atmel\Atmel Toolchain\AVR8 GCC\Native\3.4.2.1002\avr8-gnu- toolchain\avr\include\avr\iox128a1u.h			
syntax is	Atmel AVR	AVR-GCC	
different	.include "ATxmega128A1Udef.inc"	<pre>#include <avr io.h=""></avr></pre>	
	.dseg	.section .data	
	.cseg	.section .text	
	.db 1,2,3,4	.byte 1,2,3,4	
	.db "message"	.ascii "message"	
	.db "message", 0x00	.asciz "message"	
	.byte 37 ;save space for bytes	.ds.b 37	
.dw		.word	
HIGH(), LOW() hi8(), lo8(
University of Florida, EEL 4744 – File 14 © Drs. Eric Schwartz & Joshua Weaver		53	





EEL 4744 ".s" File Compilation .dseg and Data Memory Data memory is typically used to create storage of variables like Var1 and Var2 It is occasionally desired to create memory and store an initial value in that memory space, as we did for Var3 and Text The initial value is stored in program memory The .global _do_copy_data special command handles copying the data from program memory to data memory

University of Florida, EEL 4744 – File © Drs. Eric Schwartz & Joshua Weave

56

