





Optimizing Compile	rs
Optimization goal	
 generate fast code Numerous optimizations 	
complie time execution	
» constant expressions → single constant value	
 dead code elimination 	
» if statement optimization	_
 may determine that certain code won't be reachab so that code block will not be generated 	le
 e.g. two reads w/o intervening write → one of the 	m can be removed
 oops - if the read is to a device register then the and dependent conditions may actually be indep 	read values could be different endent
» multiply by power of 2 constant	
 optimized into a shift operation 	
» killers (more details next)	
 eliminate redundant memory operations 	
 reorder apparently independent memory op 	erations
» caching frequently used variables in regis	ters
 Usually good but can spell disaster 	
» when applied to device register variables	
School of Computing University of Utah 4	CS 5780



0000268);
0000268);
a) {
i this:
What is wrong?
Why did the compiler think this was OK?

















