Intrinsics

Last updated 5/14/19

Intrinsics

- ARM Assembly Instruction Intrinsics
  - High level program access to processor assembly level instructions
  - Common Intrinsics (2 leading underscores)

__delay_cycles(48_000_000); // delay 1 sec

void **__delay_cycles(** unsigned int *cycles* **);**	varies	Delays execution for the specified number of cycles. The number of cycles must be a constant. The __delay_cycles intrinsic inserts code to consume precisely the number of specified cycles with no side effects. The number of cycles delayed must be a compile-time constant. **Note:** Cycle timing is based on 0 wait states. Results vary with additional wait states. The implementation does not account for dynamic prediction. Lower delay cycle counts may be less accurate given pipeline flush behaviors.

__nop();

void **__nop(** void **);**	**NOP**	Perform an instruction that does nothing.

Intrinsics

- ARM Control Register Intrinsics
  - High level program access to the CPSR register
  - Common Intrinsics (1 leading underscore)

_enable_interrupts();

unsigned int *dst* = **_enable_interrupts( ) ;**	Cortex-M0: **MRS** *dst*, **PRIMASK** **CPSIE** *i* Cortex-M3/M4/R4/A8: **MRS** *dst*, **FAULTMASK** **CPSIE** *f*	Enable all interrupts and return previous PRIMASK or FAULTMASK setting. The assembly instructions are dependent on the architecture.

_disable_interrupts();

unsigned int *dst* = **_disable_interrupts( ) ;**	Cortex-M0: **MRS** *dst*, **PRIMASK** **CPSID** *i* Cortex-M3/M4/R4/A8: **MRS** *dst*, **FAULTMASK** **CPSID** *f*	Disable all interrupts and return previous PRIMASK or FAULTMASK setting. The assembly instructions are dependent on the architecture.

Intrinsics

- Inline Assembly Code
  - The compiler inserts your assembly language code directly as described into the compiled output.

    __asm(" assembler text "); (2 leading underscores)

  - The “assembler text” must conform to assembler code standards