

FPU: The Floating Point Unit

- *FPU*, designed to perform efficient computation with floating point numbers, is built (directly) on the Pentium processors
 1. It is backward compatible with older numerical co-processors that were provided on a separate chip (ex: 8087 up to 80387)
 2. Use the .387, .487, .587 or .687, ... to enable assembly of FPU/co-processor instructions
- There are 8 general-purpose FPU registers; each 80-bit wide

Single-precision or double-precision values of the IEEE-754 standard are placed within those 80 bits in an extended format specified by Intel. (Intel FPU's conform to the IEEE-754 standard)

General-Purpose FPU Registers

- They are organized as a stack maintained by the FPU
- The current top of the stack is referred by ST (*Stack Top*) or ST(0). ST(1) is the register just below ST, and ST(n) is the n -th register below ST
- 15 bits are reserved for the exponent: $e = E - 3FFFh$
- The "1" in the mantissa $1.f$ is stored as an explicit 1 bit at position 63. Hence f is stored from bit 0 to bit 62

	79	78	64	63	0	Tags	
ST(0)	s	Exponent		Fraction			tag(0)
ST(1)							tag(1)
ST(2)							tag(2)
ST(3)							tag(3)
ST(4)							tag(4)
ST(5)							tag(5)
ST(6)							tag(6)
ST(7)							tag(7)

The Tag Register

- The Tag register is a 16-bit register
 1. The first 2 bits, called $\text{Tag}(0)$, specify the *type* of data contained in $\text{ST}(0)$
 2. $\text{Tag}(i)$ specifies the *type* of data contained in $\text{ST}(i)$ for $i = 0, \dots, 7$
 3. The 2-bit value of $\text{Tag}(i)$ indicates the following about the content of $\text{ST}(i)$
 - 00** : $\text{ST}(i)$ contains a valid number
 - 01** : $\text{ST}(i)$ contains zero
 - 10** : $\text{ST}(i)$ contains NaN or ∞
 - 11** : $\text{ST}(i)$ is empty