Table 1–1. Custom Instruction Architectural Types, Application & Hardware Interface		
Architectural Type	**Application**	**Hardware Interface**
Combinatorial	Single clock cycle custom logic blocks	`dataa[31..0]`, `datab[31..0]`, `result[31..0]`
Multi-cycle	Multi clock cycle custom logic block of fixed or variable durations	`dataa[31..0]`, `datab[31..0]`, `result[31..0]`, `clk`, `clk_en`, `start`, `reset`, `done`
Extended	Custom logic blocks that are capable of performing multiple operations	`dataa[31..0]`, `datab[31..0]`, `result[31..0]`, `clk`, `clk_en`, `start`, `reset`, `done`, `n[7..0]`
Internal Register File	Custom logic blocks that access internal register file for input and/or output	`dataa[31..0]`, `datab[31..0]`, `result[31..0]`, `clk`, `clk_en`, `start`, `reset`, `done`, `n[7..0]`, `a[4..0]`, `readra`, `b[4..0]`, `readrb`, `c[4..0]`, `writerc`
External Interface	Custom logic blocks that interface to logic outside of the NIOS II processor's data path	Standard custom instruction signals, plus user-defined interface to external logic.