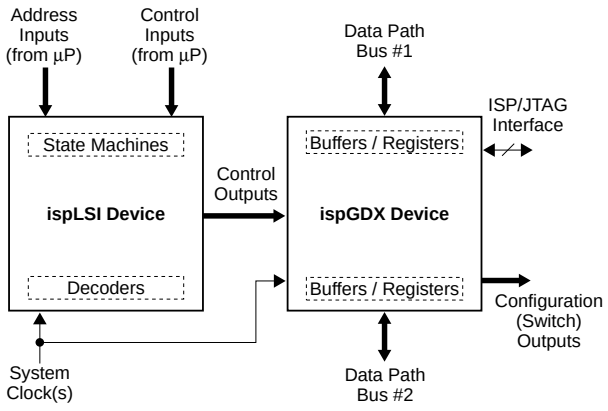
Applications

The ispGDX family architecture has been developed to deliver an in-system programmable signal routing solution with high speed and high flexibility. The devices are targeted for three similar but distinct classes of end-system applications:

Programmable, Random Signal Interconnect (PRSI)
This class includes PCB-level programmable signal routing and may be used to provide arbitrary signal swapping between chips. It opens up the possibilities of programmable system hardware. It is characterized by the need to provide a large number of 1:1 pin connections which are statically configured, i.e., the pin-to-pin paths do not need to change dynamically in response to control inputs.

Programmable Data Path (PDP)
This application area includes system data path transceiver, MUX and latch functions. With today's 32- and 64-bit microprocessor buses, but standard data path glue components still relegated primarily to eight bits, PCBs are frequently crammed with a dozen or more data path glue chips that use valuable real estate. Many of these applications consist of "on-board" bus and memory interfaces that do not require the very high drive of standard glue functions but can benefit from higher integration. Therefore, there is a need for a flexible means to integrate these on-board data path functions in an analogous way to programmable logic's solution to control logic integration. Lattice's ispLSI High-Density PLDs make an ideal control logic complement to the ispGDX in-system programmable data path devices as shown below.

Figure 2. ispGDX Complements Lattice ispLSI



Programmable Switch Replacement (PSR)
Includes solid-state replacement and integration of mechanical DIP Switch and jumper functions. Through in-system programming, pins of the ispGDX devices can be driven to HIGH or LOW logic levels to emulate the traditional device outputs. PSR functions do not require any input pin connections.

These applications actually require somewhat different silicon features. PRSI functions require that the device support arbitrary signal routing on-chip between any two pins with no routing restrictions. The routing connections are static (determined at programming time) and each input-to-output path operates independently. As a result, there is little need for dynamic signal controls (OE, clocks, etc.). Because the ispGDX device will interface with control logic outputs from other components (such as ispLSI) on the board (which frequently change late in the design process as control logic is finalized), there must be no restrictions on pin-to-pin signal routing for this type of application.

PDP functions, on the other hand, require the ability to dynamically switch signal routing (MUXing) as well as latch and tri-state output signals. As a result, the programmable interconnect is used to define *possible* signal routes that are then selected dynamically by control signals from an external MPU or control logic. These functions are usually formulated early in the conceptual design of a product. The data path requirements are driven by the microprocessor, bus and memory architecture defined for the system. This part of the design is the earliest portion of the system design frozen, and will not usually change late in the design because the result would be total system and PCB redesign. As a result, the ability to accommodate *arbitrary* any pin-to-any pin re-routing is not a strong requirement as long as the designer has the ability to define his functions with a reasonable degree of freedom initially.

As a result, the ispGDX architecture has been defined to support PSR and PRSI applications (including bidirectional paths) with no restrictions, while PDP applications (using dynamic MUXing) are supported with a minimal number of restrictions as described below. In this way, speed and cost can be optimized and the devices can still support the system designer's needs.

The following diagrams illustrate several ispGDX applications.