1.4 System Controller Operations

The VIC068A is able to assume the system controller functions (also known as slot 1 functions) by strapping the SCON* signal Low. For reliable operation, the SCON* signal must remain asserted for the duration of operation. As the system controller, the VIC068A performs the following functions:

- priority, round robin, or single-level arbitration
- driving IACK* daisy-chain
- driving BGiOUT* daisy-chain (all four levels)
- driving SYSCLK output
- driving SYSRESET* output
- driving BCLR*
- VMEbus arbitration timeout timer

The following VIC068A registers are used as the system controller:

- Transfer Timeout Register (TTR), bits 5−7
- Arbiter/Requestor Control Register (ARCR), bit 7
- Error Group Interrupt Control Register (EGICR), bit 5

1.4.1 VMEbus Arbitration

The arbitration scheme is programmed by writing ARCR[7]. In PRI(priority) mode, BR3* has the highest priority and BR0* has the lowest. Higher priority bus requests will be handled before lower priority bus requests when in PRI mode. In the RRS(round robin) scheme, arbitration priority is assigned on a rotating basis. When the bus is granted to a requester on bus request line BR[n]*, then the highest priority for the next arbitration is assigned to bus request line BR[n−1]* (or BR3* if previous level was BR0*). Single-level arbitration is obtained by programming the VIC068A for PRI and setting all requestors to the same level.

When the VIC068A is system controller, it senses the state of the BRi* inputs. One of the four BGiOUT* signals is asserted, corresponding to the highest pending request level dur-

ing that arbitration cycle. If the VIC068A, as system controller, has a BRi* pending along with another potential master at the same request level, the VIC068A does not assert the BGiOUT* for itself.

An arbitration cycle begins with the deassertion of the BBSY* signal. The VIC068A waits a minimum of 3T after the deassertion of BBSY* before asserting the BGiOUT* signal. The VIC068A deasserts the BGiOUT* signal when the BBSY* is again reasserted.

The VIC068A asserts the BCLR* signal as part of its arbiter function when it senses a request at a higher priority than the level of the current VMEbus master. This may occur when the VIC068A is enabled for both PRI and RRS arbitration schemes. In either case, the VIC068A deasserts BCLR* when BBSY* is deasserted.

In systems containing many contending VMEbus masters, the use of RRS arbitration and fair requests is strongly recommended to prevent excessive bus latency to some of the VMEbus masters. To allocate an unequal share of bus bandwidth to a particular master, assign that master to a BR* level shared with fewer masters.

1.4.2 The VMEbus Arbitration Timeout Timer

After the VIC068A has asserted the BGiOUT* signal, the VIC068A system controller monitors how long the grant is active. Failure to assert BBSY* within 8 µs causes the VIC068A to issue its own BBSY* for the VMEbus-required 90 ns. The EGICR can be used to generate an interrupt for a VMEbus arbitration timeout condition. This timeout feature may not be disabled. See section 1.9.5.

1.4.3 The VMEbus Transfer Timeout Timer

The VIC068A contains a VMEbus transfer timeout timer. When the VIC068A is configured as the system controller, and the transfer timeout timer is enabled, the VIC068A starts this timer at the assertion of a DSi*. If the timer expires before the assertion of DTACK* or BERR*, BERR* is asserted by the system controller. BERR* remains asserted until the DSi*s are removed. The timer is configured in the TTR[7:5]. BESR[4] is set when this timeout condition occurs.

1.4.4 The BGi Daisy-Chain Driver

The VIC068A, as system controller, drives the BGiOUT* daisy-chain in response to VMEbus requests. When the VIC068A is the system controller, the BGiIN* lines are inactive, but need to be pulled High externally at the VIC068A (4.7 − 10KΩ).

1.4.5 The IACK* Daisy-Chain Driver

The VIC068A, as system controller, is the first device to drive the IACK* daisy chain (*Figure 1−5*). When the VIC068A is performing duties as the system controller, the IACK* input is internally tied to the IACKIN* input. When a VMEbus interrupt handler drives IACK* Low on the VMEbus, the system controller VIC068A will see this as a Low on its IACK* input and will react just like a VIC068A located elsewhere on the VMEbus would when its IACKIN* is driven Low. See section 1.9.2.

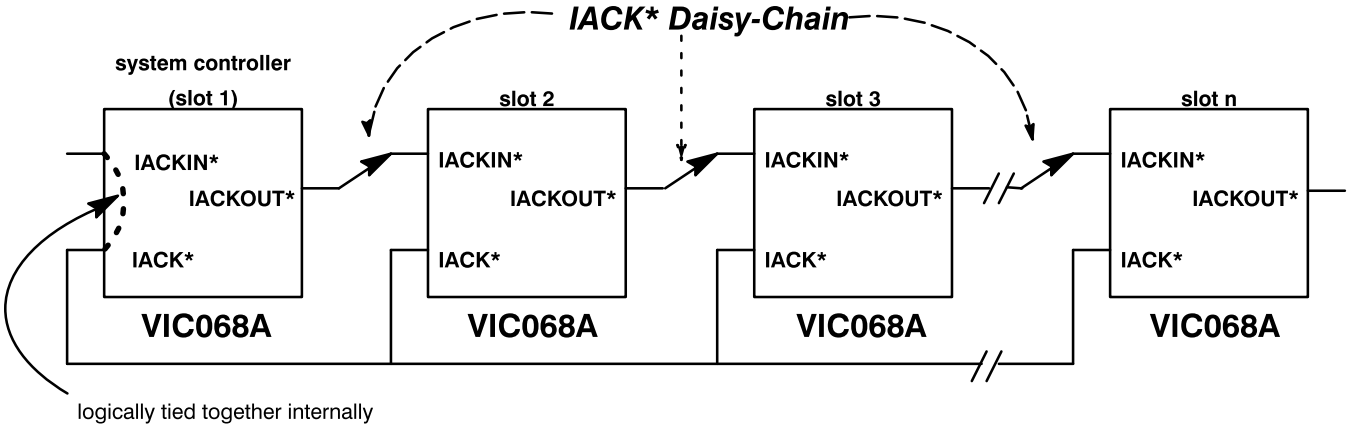

Figure 1−5. IACK* Daisy-Chain