Coprocessor Expansion and 86 Pin Signals on Amiga Computers

by Dave Haynie

June 24, 1987

ABSTRACT

This document details the signals found on the various types of 86 pin expansion connectors on different Amiga computers, especially the signals found on the B2000 computer's 86 pin coprocessor slot, and how these differ from the similar signals found on A2000 computers and those of the original A1000 computers. This paper also explains the Coprocessor Slot's autoconfiguration and DMA protocols and how they fix the problems introduced in the A2000 Coprocessor Slot.

1. Introduction

This document details the signals found on the various types of 86 pin expansion connectors on different Amiga computers, especially the signals found on the B2000 computer's 86 pin Coprocessor Slot, and how these differ from the similar signals found on A2000 computers and those of the original A1000 computers. This paper also explains the Coprocessor Slot's autoconfiguration and DMA protocols and how they fix the problems introduced in the A2000 Coprocessor Slot.

1.1. Changes From Previous Documents

We've kept the 86 pin specification on the B2000 as similar to those available on the A2000, A1000, and A500, wherever possible. However, some major changes were absolutely required. With the design of the A2000, the function of the 86 pin slot had shifted from general expansion connector to expansion specifically intended for coprocessors and similar devices. Thus, while the A500's and A1000's 86 pin connectors have to support both some kind of coprocessor expansion and the normal ZORRO expansion, the A2000 machines can optimize each slot for its purpose if required. Or if necessary, which is more the case.

The 86 pin connector on the A500 and A1000 becomes something of an advantage, because of the fact that all expansion must be done externally. When a coprocessor device (something that needs to completely replace the 68000 in all forms of bus access and operation, like a 68020 accelerator card) is added, it can physically sit between the computer motherboard and the 100 pin expansion box, thus allowing the device to completely replace the action of the motherboard's processor from the point of view of the expansion box. A machine with both slots on the motherboard must provide some facility to logically insert the 86 pin slot in front of the 100 pin slot for certain applications[1].

In the A2000, the Coprocessor Slot signals that control DMA can be used to insert the coprocessor in the place of the normal 68000 via the standard 68000 DMA request protocol. This, however, isn't a totally transparent replacement; the action of the coprocessor taking control over the local bus from the 68000, in the A2000, can block other DMA events coming over from the 100 Expansion Bus. For total control of the Expansion Bus on the A2000, the 68000 could be physically removed from the motherboard, but that would result in the "coprocessor" being a complete "replacement" processor, with no swapping between the two permissible. The B2000 solves these problems with a higher-level DMA protocol between the main and coprocessor devices. This is covered in a later chapter.

1.2. Definition of Terms

Several terms are used in the following text, and an understanding of them is required to speak proper Amigaese. A device known as a PIC, or Plug In Card, is a device that plugs into an Expansion Slot and follows the autoconfiguration protocol. Nothing should plug into a 100 pin slot that doesn't follow this protocol. The term slot refers to a physical plug in location, either the Coprocessor Slot or one of the five available Expansion Slots. The terms 100 Pin Slot and Expansion Slot are considered synonyms, and describe one of the five 100 pin Expansion Slots. The Expansion Bus is the processor bus that is in common between all Expansion Slots. The terms 86 Pin Slot, Coprocessor Slot, and Local Slot are considered synonyms, and pertain to the 86 pin Coprocessor Slot in the A2000 and B2000. The terms 86 Pin Edge and Expansion Edge are considered synonyms, and pertain to the 86 pin Expansion Edge in the A1000 and A500. The Local Bus is the processor bus directly connected to the 68000 processor and the Coprocessor Slot or Expansion Edge; both the Coprocessor Slot and Expansion Edge are considered Local Bus Ports. Each different implementation of a hardware design is termed an Instance of that design; thus, the A2000's Coprocessor Slot, the B2000's Coprocessor Slot, and the A500 and A1000 Expansion Edges are instances of Local Bus ports.

Along with an understanding of Amiga bus terms, a familiarity with Motorola's 68000 processor and its characteristic names for things will also be very useful in understanding this document.

[1] Please refer to the paper "100 Pin Signals on Amiga Computers" by Dave Haynie, for a complete discussion of the 100 pin bus.

2. Coprocessor Slot Signals

The Coprocessor Slot signals discussed below apply for most of the machines, though in some cases the item mentioned exists on only some of the machines; this will be specifically. Most of these signals are directly in common with the 68000, or directly a part of the 68000 local bus, instead of being buffered as on the Expansion Bus. No signal on a Coprocessor card should load the Local Bus with more than one "F" series standard load.

2.1. Power Connections

The Coprocessor Slot provides several different voltages designed to supply Coprocessor devices. The A2000 power supply is currently rated at 200 Watts, which supplies the main board and all other expansion ports as well as the Coprocessor Slot.

2.1.1. Digital Ground (Ground)

Digital supply ground used by all expansion cards as the return path for all expansion supplies. This are found on all instances of the Local Bus ports. See Appendix for pin assignments.

2.1.2. Main Supply (+5V)

Main power supply the Coprocessor slot, and is can supply up to 2.0 Amps of +5VDC on the A2000. The maximum supply current for the entire A2000 system is 20 Amps for all devices inside the A2000 that use +5V, including the motherboard. The corresponding pins on the Expansion Edge of the A1000 can source only 1 Amp, and even less on the A500. Pins: 5, 6.

2.1.3. Negative Supply (-5V)

Negative version of the main supply, for small current loads only; there’s a total of 0.3 Amp for the entire A2000 system. This pin is similar to what’s available on the A1000 and A500, though these other instances will have different currents available. Pin: 8.

2.1.4. High Voltage Supply (+12V)

Higher voltage supply, useful for communications cards an other devices requiring greater that digital voltage levels. This is intended for small loading only; there’s a total of 8 Amps for the entire A2000 system, much of which is normally devoted to floppy and hard disk drive motors. Found on all instances of Local Bus Ports, though the available currents may vary. Pin: 10.

2.2. Clock Signals

There are various system clocks available at all Local Bus Ports, useful in designing synchronous Coprocessor systems. Loading on these clocks should be watched very carefully on all types of Amiga computers.

2.2.1. /C1 Clock

This is a 3.58 MHz clock (3.55 MHz on PAL systems) that’s synched to the falling edge of the 7M system clock. Also known as /CCK in some places. Pin 16.

2.2.2. /C3 Clock

This is a 3.58 MHz clock (3.55 MHz on PAL systems) that’s synched to the rising edge of the 7M system clock. Also known as /CCKQ in some places. Pin 14.

2.2.3. CDAC Clock

This is a 7.16 MHz clock (7.09 MHz on PAL systems) that leads the 7M system clock by about 70ns (90 degrees). Pin 15.

2.2.4. E Clock

This is the 68000 generated "E" clock, used for 6800 family peripherals driven by "E" and 6502 peripherals driven by PHI2. This clock is six 7M clocks high, four clocks low, as per the 68000 spec. This clock is always generated by the 68000, regardless of the state of the bus and the Coprocessor; this fact should be considered by the Coprocessor implementor when designing any Coprocessor /VMA logic. Pin 50.

2.2.5. 7M Clock

This is the 7.16 MHz system clock (7.09 MHz on PAL systems). This is available only on the B2000 at this pin, and is in common with the 68000's clock input. This pin, pin 7, is unused on all other Local Bus Port instances. Many applications that run on systems without the 7M clock create a 7M equivalent clock, using the relationship EQU7MHz = /C1 XNOR /C3; care must be taken in considering any additional delays that this equivalent clock causes on systems other that the B2000.

2.2.6. 28MHz Clock

This is the 28.64 MHz fundamental clock used to derive all other system clocks under normal operation. There's no guaranteed phase relationship between this clock and the system clocks. When the system is being driven by an external clock source via XCLK and /XCLKEN, this clock will essentially be completely asynchronous to the system clocks. It is provided mainly to provide a fast clock for fast coprocessors. This is pin 9 on the Coprocessor Slot, and is an unused pin on the Expansion Edge of the A500 and A1000.

2.3. Addressing and Control Signals

These signals are various items used for the addressing of resources on a coprocessor card by the by the 68000 and any DMA devices, and for 24 by 16 bit addressing of other system resources by a coprocessor device (which may easily have more potential). Most of these signals are directly in common with 68000 signals.

2.3.1. Read-Write (R/W)

The 68000's R/W output. When driven high it indicates a read or internal cycle, when driven low it indicates a write cycle. When the coprocessor takes over it drives this line; the 68000's output will tri-state. Pin 68.

2.3.2. Address Bus (A1-A23)

This directly connects to the 68000's address bus, providing 16 megabytes of address space with 23 bits of address for a 16 bit data bus. The 68000 is capable of driving only this much address space. Thus, any resources on a coprocessor board must map somewhere into the 68000 memory space. The best thing to do with any such memory is allow it to be autoconfigured by the 1.2 OS; this will place it somewhere in the 8 megabyte space starting at $200000 (the A2000 doesn't support autoconfiguration from the Coprocessor Slot, the B2000 does). Any resources intended specifically for the coprocessor only can be located above the 68000's 16 megabyte space if the coprocessor hardware permits that extended addressing. All board and Expansion bus resources will normally map into the first 16 megabytes of the address space of a coprocessor board. See Appendix for pin list.

2.3.3. Address Strobe (/AS)

The falling edge of this strobe indicates that addresses are valid, the rising edge signals the end of the memory cycle. This is in common with the 68000 /AS signal. The coprocessor drives this signal when it takes over; the 68000's will tri-state. Found on pin 74.

2.3.4. Data Bus (D0-D15)

This is directly connected to the 68000's data bus, providing 16 bits of data accessible by word or either byte. Any coprocessor handling words larger than 16 bits must either step down to 16 bits on its own or provide circuitry to convert the 16 bit word size of the main board and Expansion Bus to the natural size of such a coprocessor, when accessing main board resources. See Appendix for pin list.

2.3.5. Data Strobes (/LDS, /UDS)

This these are the 68000's upper and lower data strobes. The strobes fall on data valid during transfer; the lower strobe being used for the lower byte (even byte address), the upper strobe being used for the upper byte (odd byte address). Like /AS, these must be driven by the Coprocessor as it assumes control, as the 68000 pins will tri-state. Pins: 70, 72.

2.3.6. Valid Memory Address (/VMA)

Output from the 68000 indicating a valid address for 6800 style peripheral devices, in response to a /VPA input. This output goes tri-state when the Coprocessor takes over from the 68000, and as such must be re-created by the coprocessor in response to a VPA signal from somewhere on the motherboard. Pin 51.

2.3.7. Valid Peripheral Address (/VPA)

Input to the 68000 indicating the address has selected a 6800 or 6502 style peripheral, so the 6800 style peripheral access should take place. When the 68000 has given up the bus to the Coprocessor, this input is ignored and must be handled by the Coprocessor board. Pin 48.

2.3.8. Data Transfer Acknowledge (/DTACK)

This signal is the 68000's Data Transfer Acknowledge input, though its being driven on the motherboard under most conditions. Normally in the Amiga system, Amiga system logic creates /DTACK for a simple, no-wait state cycle (this may be varied by the custom chips). Therefore, this signal is treated as an output to the Expansion and Coprocessor Slots, for most situations. Any slow device on the bus that needs to control /DTACK may do so by negating XRDY to hold off /DTACK or asserting /OVR very quickly to tri-state /DTACK. Any any coprocessor must be able to support this action by Expansion boards as well. Note that depending upon when /AS is asserted by a bus master when accessing the CHIP memory, one of two possible cycles may result. If /AS is asserted during C1 low, C3 low, the bus cycle is considered "in-sync", and will proceed, with /DTACK driven as for a normal, 4 tick clock cycle. If instead, /AS is asserted during C1 high, C3 high, the bus cycle is considered "out of sync" and the internally generated /DTACK will be held off, causing a wait state that's designed to "sync-up" the DMA cycle with the custom chip's memory cycle. Of course, when a coprocessor is accessing any of its on-board resources the designer can implement any data transfer reasonable scheme that comes to mind. This signal is on pin 66.

2.3.9. Processor Status (FC0-FC2)

These signals are the 68000 Processor Status outputs, which can be used by bus devices to determine the internal state of the 68000 any time /AS is asserted. When a coprocessor is in charge, it must drive these pins in a way compatible with how the 68000 does it. The different 68000 status codes can be found in any 68000 spec sheet. Pins 31, 33, 35.

2.3.10. Bus Error (/BEER)

This is an input that goes directly to the 68000. Its used to indicate the occurrence of some kind of bus error. Any Expansion Card capable of detecting a bus error relating directly to that card can assert /BEER when that bus error condition is detected. At other times, the card must monitor /BEER and be prepared to tri-state all of its on-bus output buffers whenever this signal is asserted. The Coprocessor card won't have to tri-state on /BEER, but it must note it and provide some way of handling the occurrence (the 68000 under normal Amiga OS control merely signals an Guru Error based on the Bus Error Exception). Since any number of devices may assert /BEER, and nearly everything in the system must monitor it, any device that drives /BEER must drive with an open collector or similar device capable of sinking at least 12ma, and any device that monitors /BEER should place as little load on it as possible (1 "F" type load or less, per board, is suggested). This signal is connected to a low valued on-board pullup resistor, and shouldn't need any more pulling up. Pin 46.

2.3.11. System Reset (/RST)

Pin 53 of the bus contains the /RST signal which is in common with the original 68000 reset signal. The /RST signal is bidirectional, and the 68000 tri-states it when the coprocessor takes over. It is only necessary for the processor to output this signal if it needs to reset the system under program control. The /RST signal is connected to a medium valued on-board pullup resistor and shouldn't need any more pulling up. The coprocessor must monitor this signal and respond to it appropriately; this may mean a complete reset, but it doesn't have to. The Coprocessor can also assert this line if a system reset is desired. up.

2.3.12. System Halt (/HLT)

This is the 68000's processor halt signal, tied directly to the 68000. It is connected to a medium valued on-board pullup resistor and shouldn't need any more pulling up. This signal, when asserted, will halt and tri-state the 68000 at the end of the current bus cycle. If driven by the 68000, it indicates detection of a double bus fault. For a complete system reset, the 68000 looks for both the /RST and /HLT lines to be asserted. The Coprocessor should handle this signal in a similar fashion. Pin 55.

2.3.13. Decoded Interrupts

Two of the 68000 non-encoded interrupt inputs are available at the Coprocessor slot, on pin 19 for interrupt level 2 (/INT2) and on pin 22 for interrupt level 6 (/INT6). These are the same interrupts used by the Amiga internal system chips and encoded by the Paula chip. They can be used by a Coprocessor board by driving them to generate 68000 interrupts when the 68000 is in charge, though generally they don't do much when the Coprocessor is in charge.

2.3.14. Encoded Interrupts (/IPL0-/IPL2)

The Coprocessor Slot provides the encoded interrupt lines /IPL0, /IPL1, and /IPL2 on bus pins 40, 42, and 44 respectively, which are the normal encoded interrupt inputs to the 68000. Nothing on the Coprocessor slot can drive these lines, but they must be monitored by any Coprocessor or alternate processor that needs to be able to respond to any system interrupts when acting as the bus master.

2.3.15. Override (/OVR)

The /OVR, or Override, signal is a special Amiga expansion signal that can serve two purposes. The signal can basically turn off the on-board decoding of system memory ranges, including those used by the Amiga custom chips. As a result of this, it can also turn off internally generated things, like /DTACK. The timing in the A500 and B2000, based on the Gary chip (not the PALs of the older machines), effectively prohibits the use of /OVR for the area of address space outside of the $200000-$9FFFFF (Expansion Bus) range.

The supported use of this signal is to allow the Coprocessor board to create its own /DTACK. Asserting /OVR will tri-state the motherboard generated /DTACK signal, allowing it to be driven. The same effect can be achieved for most applications by using XRDY to delay the motherboard's generation of /DTACK. Pin 17. Pin 17.

2.3.16. External Ready (XRDY)

This input provides a way for an external device to delay the motherboard generated /DTACK, for things like slow memory and I/O boards that need to add wait states. This signal should be negated very quickly, no later than 60ns from address valid (/AS asserted), in order for the motherboard circuitry to have enough time to prevent the normal assertion of /DTACK. XDRY should stay negated for as many wait states are required. Once XRDY is asserted, /DTACK completes the rest of the normal cycle. XRDY is a wired-OR input; it is pulled up by a resistor on the motherboard, and should be driven with an open collector or equivalent output. Pin 18.

2.3.17. Configuration Chain (/COPCFG)

Pins 11 and 12 are basically the configuration IN and configuration OUT signals. Pin 12, the configuration IN input, is grounded on all versions of the Local Bus Ports, indicating that this Slot is the first in any configuration chain and may proceed with configuration. On the A500, A1000, and A2000, the configuration OUT signal, pin 12, is a no-connect. Because of this, its impossible to normally autoconfigure any device in the Coprocessor slot of an A2000. On the B2000, pin 11 is a true configuration OUT signal, which becomes the configuration IN input to the first Expansion Slot. Thus, the coprocessor slot is configured first on the B2000. A note of caution here, though. All normal Expansion Bus devices assert their /SLAVE output whenever they respond to an address. This /SLAVE output allows the collision detect circuitry to determine if multiple devices are responding to the same address. When a collision is detected this way, the /BEER signal is asserted, causing all PICs to tri-state, and saving both these PICs and the Expansion Bus drivers from any potentially destructive buffer fights. While the Coprocessor slot on the B2000 can be automatically configured, it can't assert a SLAVE signal for collision detect. Thus, designers must be very careful with any autoconfiguring resources on a Coprocessor card.

During the autoconfiguration process, first the Coprocessor card, then all an unconfigured PICs in turn, respond to the 64K address space starting at $E80000 as their respective CFGIN signals are asserted. All unconfigured PICs come up with CFGOUT negated. When configured, or told to "shut up", the Coprocessor Card or any PIC should assert CFGOUT, which results in the CFGIN of the next slot to be asserted. On-board logic automatically passes on the state of the previous CFGOUT to the next CFGIN for any slot not occupied by a PIC, so there's no need to sequentially populate the Expansion Bus Slots and no need to have the Coprocessor Card do any autoconfiguring if there's no need for real autoconfiguration.

2.4. DMA and Coprocessor Signals

This will be covered in more detail in the next chapter, but this section covers the basic signals involved in DMAs and the Coprocessor interface.

2.4.1. Bus Request (/BR, /CBR)

All instances of Local Expansion Ports have a Bus Request to 68000 of some kind. In the A2000, as in the A500 and A1000, this is directly connected to the 68000's /BR input, which is considered a wired-OR input; all devices driving this input must technically drive it with an open collector or equivalent driver. In actuality, the A500 and A1000 don't use this at all internally, so a standard driver may be used if necessary. The A2000's /BR input is shared by the /BR output of the DMA arbitration logic, so this will be necessary on an A2000 Coprocessor Slot device. The B2000 has in place of the 68000's /BR line a special bus request all its own, /CBR. The difference between this and /BR is covered in the next chapter. In both cases, the signal is an input to the 68000 used to request mastership of the Local Bus. The signal is found on pin 60.

2.4.2. Bus Grant (/BG, /CBG)

All instances of Local Expansion Ports have a Bus Grant from the 68000 of some kind. In the A2000, as in the A500 and A1000, this is directly connected to the 68000's /BG output. In the B2000, a Coprocessor specific Bus Grant signal, /CBG, is in its place. In either case, the signal is asserted by the 68000 in response to a Bus Request. This indicates to the device in the Coprocessor slot that the 68000 will fully relinquish the bus at the end of this cycle. A /BG received on the Coprocessor Slot in an A2000 could be a Grant given in response to an Expansion Bus DMA request as well as one in response to the Coprocessor Slot DMA request. On the B2000, /CBG will **only** be asserted if the Coprocessor Slot is granted the bus. This signal is found on pin 64.

2.4.3. Bus Grant Acknowledge (/BGACK)

This is the 68000's /BGACK, or Bus Grant Acknowledge, signal. Any device that receives a bus grant from the 68000 should assert this signal as long as the DMA continues, releasing it once the DMA request is finished. This signal should never be asserted until the specific Bus Grant has been received, /AS is negated, /DTACK is negated, and /BGACK itself is negated, indicating that all other potential bus masters have relinquished the bus. This output is driven as a wired-OR output, so all devices driving it must drive it with an open collector or equivalent device. Pin 62.

2.4.4. Coprocessor Grant Acknowledge (/BOSS)

This signal exists only on the B2000, on pin 20. That pin is unused on both the A2000 and the A500. Originally, this pin was called /PALOPE on the A1000, and was part of the planned ROM expansion method. This is currently obsolete; the method of ROM expansion was changed to work without the need for such a signal. On the B2000, the /BOSS signal is driven by a Coprocessor instead of /BGACK when the Coprocessor wishes the DMA access granted it to be a true Coprocessor access, not a simple DMA. This is all explained in the following chapter.

3. The B2000 Coprocessor Interface

The B2000 computer implements an extended version of the A2000's Coprocessor Slot, designed to make the swapping of main processors under program control much more powerful and transparent to the rest of the B2000 system. There are things that can be done from the B2000 Coprocessor Slot that can't be done from the A2000's Coprocessor Slot, so this is an important consideration to anyone designing a Coprocessor device of some kind.

3.1. Normal 68000 DMA Architecture

The 68000 supports hardware signals designed to permit a simple DMA protocol. This protocol allows multiple devices to take control of the 68000's data, address, and control buses. When a device of some kind desires direct access to the 68000's bus, it asserts the /BR (Bus Request) input of the 68000. Once /BR is asserted, the 68000 will complete whatever operation its doing to the point at which it can cleanly relinquish its bus. At this point, it will assert its /BG (Bus Grant) output, telling the device requesting DMA that its just about ready to shut down. The requesting device then issues /BGACK (Bus Grant Acknowledge) as soon as the 68000 is completely off the bus (/DTACK and /AS are negated). When the DMAing device is done with the bus, it releases /DTACK and /BR, and the 68000 will then release /BG.

3.2. Where the 68000 DMA Protocol Fails

The above protocol, as implemented in the 68000, is sufficient for many type of DMA operation, especially for simple things in which there are single DMA devices on the bus. What this doesn't easily account for are multiple DMA devices. While the /BR and /BGACK inputs to the 68000 can be wire-ORed to support several devices, there are still problems with this scheme. Should multiple devices request DMA at the same time, the 68000 will see nothing different that if only one device is requesting DMA. While careful monitoring of the /BGACK by responding potential bus masters can solve some of the problems, there are much cleaner approaches to this problem.

One such solution is implemented in the ZORRO and A2000/B2000 Expansion Buses. Each slot on the Expansion Bus has its own private Bus Request and Bus Grant. Each Bus Request signal is considered by a priority encoding and latching circuit. The result of this is that, if simultaneous Bus Requests come in from Expansion Slots, only the Slot given higher priority will actually get a Bus Grant. Any Bus Requests that come in while another DMA is in effect will be held off until the 68000's /BG line has been negated for at least one tick. This circuitry, part of the original ZORRO specification, eliminates the problems that can occur with various DMA devices all competing for the Expansion and Local buses.

3.3. Where the Expansion DMA Protocol Fails

With the introduction of the A2000 and its Coprocessor Slot, two new problems have appeared. One of these is the old priority problem. All of the Expansion Bus Requests are priority encoded into one signal that drives the 68000 /BG input. With any device in the A2000 Coprocessor Slot that desires the Local Bus, however, the old conflict is going to reappear. For now we've got two distinct, unsynchronized potential bus masters, the Coprocessor Slot and the whole encoded Expansion Bus. If the Coprocessor Slot and the Expansion Slot each request the bus at the same time, one Bus Grant, on /BG from the 68000, will be sent back to both devices, and then its a race, and possibly a messy tie, to see who gets to assert /BGACK first.

The next problem is Coprocessor specific. Lets assume for the moment that a Coprocessor device has successfully requested the bus from the 68000. To simplify things, lets assume that its a 68020 processor board, running the Amiga OS, content to never let the 68000 back on. It goes romping' along, with /BR, /BG, and /BGACK each asserted to keep the bus to itself. Now, along comes another device, like a shiny new DMA Hard Disk Controller Card sitting in an Expansion Slot. It wants to DMA something to memory, as you'd expect. But lo and behold, this 68020 card has the bus tied up; its got all the bus arbitration signals asserted, blocking any other device from taking over the bus.

So in this setup, there are three options. The first, of course, is to keep the 68020 on all the time, ignore your DMA Hard Disk, and just run with floppies and non-DMA devices. That's no solution, so we'll throw that one out. Next, we can have the Coprocessor give up the bus whenever someone else wants a DMA, letting the 68000 on for a second, which in turn lets the disk controller have the bus, the DMA takes place, the 68000 goes back on for just long enough to let the Coprocessor take over. This could work, but first of all its sloppy and forces DMAs to take place in twice the time they would without the Coprocessor, and in the second place this requires a new DMA protocol to be observed by all the Expansion Bus DMA devices anyway, since they have to have some way of letting the Coprocessor know that they want the bus. This is no good, Expansion Bus things shouldn't care which processor is running the local bus. As a modification to this method, the Coprocessor could shut up every so often, letting the 68000 on the bus just to see if any DMAs are pending. But that's a performance hit too, and shouldn't be necessary. The final option with the A2000 setup is to physically remove the 68000 from the motherboard, allowing the Coprocessor to plug in as a complete replacement, bypassing the need for the Coprocessor to request the Local Bus -- it is completely given the Local Bus. The problem with this last scheme is that now the Coprocessor is no longer a Coprocessor, but a complete replacement processor. There's no way for the Coprocessor and 68000 to function together in such a system. This may not be important some Coprocessing applications, but it removes the possibility of some powerful systems.

3.4. The B2000 Coprocessor Solution

The B2000 hardware has implemented a more sophisticated Coprocessor system that removes these problems. The B2000 Coprocessor Slot has a signal called /CBR (Coprocessor Bus Request) as a replacement for /BR, a signal called /CBG (Coprocessor Bus Grant) as a replacement for /BG, and one additional signal, /BOSS, which is also known as Coprocessor Grant Acknowledge.

Under the B2000 system, there are essentially two ways a Coprocessor devices can receive a Local Bus mastership. Both start in the same way. To request the bus, the Coprocessor asserts /CBR. Instead of going directly to the 68000, this signal is prioritized and latched along with any Expansion Slot /BR signals. The /CBR signal has the highest DMA priority. Assuming no other DMAs are currently active, the 68000 will issue a Bus Grant via /BG, which will go to the prioritizer[2] and result in /CBG being asserted. At this point, all other DMA requests will be locked out; no other /BGs of any kind will be issued. Following the normal 68000 protocol at this point, the Coprocessor will assert /BGACK when the 68000 is off the bus, and it will have bus access as before. And as before, it is holding off any further DMAs from the Expansion Bus. Which may be what was wanted; this type of DMA access is very similar to what a normal DMA device from the Expansion Bus would achieve.

There is another way to take over the Bus, however. This starts the same as before, with a /CBR resulting in a /CBG. One the Coprocessor has received its Bus Grant, however, it does something different. What it does is assert the /BOSS signal instead of /BGACK. This has several immediate effects. First of all, the 68000 sees /BOSS as the same thing as /BGACK, so it stays off the bus just as if /BGACK had been asserted. Next, the data direction of /CBR and /CBG change on the Coprocessor Bus. The /CBR signal is now an output from the bus control logic, the prioritized and latched combination of all the /BR signals from the Expansion Bus. The /CBG signal is now an input going into the bus control logic that will be passed on to the Expansion Bus in response to an Expansion Bus /BR. The bus control logic also holds /BR to the 68000 in a low state. The data direction of /CBR and /CBG changes with a change in /BOSS, so the lines that alternately drive /CBR and /CBG on a Coprocessor card should be enabled and disabled with the assertion of /BOSS.

Anyway, what all this means is that, in asserting /BOSS instead of /BGACK, the Coprocessor has the bus, the 68000 is in tri-state, and any of the Expansion Slots may initiate a DMA of the Coprocessor at any time, directly, according to the normal /BR -> /BG -> /BGACK protocol of the 68000. The Coprocessor can allow the 68000 back on the bus by negating the /BOSS line. Thus, the Coprocessor can be a real Coprocessor, functioning as the equivalent of the 68000 for all things as far as the whole Amiga system is concerned.

4. Appendix

Here are the four instances of the 86 pin Local Bus, the A500 and A1000 Edge connectors, used for all kinds of expansion on those machines, and the A2000 and B2000 Coprocessor slots.

[2] The B2000 system does all of its DMA prioritization via the "Buster" custom bus controller chip.

PIN	A500	A1000	A2000	B2000	Function
1	X	X	X	X	Ground
2	X	X	X	X	Ground
3	X	X	X	X	Ground
4	X	X	X	X	Ground
5	X	X	X	X	+5VDC
6	X	X	X	X	+5VDC
7	X	X	X		No Connect
				X	7M Clock
8	X	X	X	X	-5VDC
9	X	X			No Connect
			X	X	28MHz Clock
10	X	X	X	X	+12VDC
11	X	X	X		No Connect
				X	/COPCFG (Configuration Out)
12	X	X	X	X	CONFIG IN, Grounded
13	X	X	X	X	Ground
14	X	X	X	X	/C3 Clock
15	X	X	X	X	CDAC Clock
16	X	X	X	X	/C1 Clock
17	X	X	X	X	/OVR
18	X	X	X	X	XRDY
19	X	X	X	X	/INT2
20		X			/PALOPE
	X		X		No Connect
				X	/BOSS
21	X	X	X	X	A5
22	X	X	X	X	/INT6
23	X	X	X	X	A6
24	X	X	X	X	A4
25	X	X	X	X	Ground
26	X	X	X	X	A3
27	X	X	X	X	A2
28	X	X	X	X	A7
29	X	X	X	X	A1
30	X	X	X	X	A8
31	X	X	X	X	FC0
32	X	X	X	X	A9
33	X	X	X	X	FC1
34	X	X	X	X	A10
35	X	X	X	X	FC2
36	X	X	X	X	A11
37	X	X	X	X	Ground
38	X	X	X	X	A12
39	X	X	X	X	A13
40	X	X	X	X	/IPL0
41	X	X	X	X	A14
42	X	X	X	X	/IPL1
43	X	X	X	X	A15
44	X	X	X	X	/IPL2

45	X	X	X	X	A16
46	X	X	X	X	/BEER
47	X	X	X	X	A17
48	X	X	X	X	/VPA
49	X	X	X	X	Ground
50	X	X	X	X	E Clock
51	X	X	X	X	/VMA
52	X	X	X	X	A18
53	X	X	X	X	/RST
54	X	X	X	X	A19
55	X	X	X	X	/HLT
56	X	X	X	X	A20
57	X	X	X	X	A22
58	X	X	X	X	A21
59	X	X	X	X	A23
60	X	X	X		/BR
				X	/CBR
61	X	X	X	X	Ground
62	X	X	X	X	/BGACK
63	X	X	X	X	D15
64	X	X	X		/BG
				X	/CBG
65	X	X	X	X	D14
66	X	X	X	X	/DTACK
67	X	X	X	X	D13
68	X	X	X	X	R/W
69	X	X	X	X	D12
70	X	X	X	X	/LDS
71	X	X	X	X	D11
72	X	X	X	X	/UDS
73	X	X	X	X	Ground
74	X	X	X	X	/AS
75	X	X	X	X	D0
76	X	X	X	X	D10
77	X	X	X	X	D1
78	X	X	X	X	D9
79	X	X	X	X	D2
80	X	X	X	X	D8
81	X	X	X	X	D3
82	X	X	X	X	D7
83	X	X	X	X	D4
84	X	X	X	X	D6
85	X	X	X	X	Ground
86	X	X	X	X	D5