SIEMENS

Microcontrollers

ApNote AP1619

☐ additional file
APXXXX01.EXE available

C167CR - Specification Update

Master/Slave Bus Arbitration

K.H. Mattheis / Siemens HL MCB PD

1 Master/Slave Bus Arbitration 3

2 Initialization 5

2.1 Master Initialization 5

2.2 Slave Initialization 6

3 Operation of the master/slave bus arbitration 6

AP1619 ApNote - Revision History		
Actual Revision : Rel.01 Previous Revison: none		
Page of actual Rel.	Page of prev. Rel.	Subjects changes since last release)

1 Master/Slave Bus Arbitration

In order to allow an easy connection of two C167 to the same external bus for sharing the same external memory resources, the bus arbitration operation is enhanced to allow the C167CR to act as a slave.

Figure 1 shows such an application with a shared memory system between the master and the slave controller. In addition, it is possible for both controllers to share internal XBUS peripherals (if existing) to the other partner in the system.

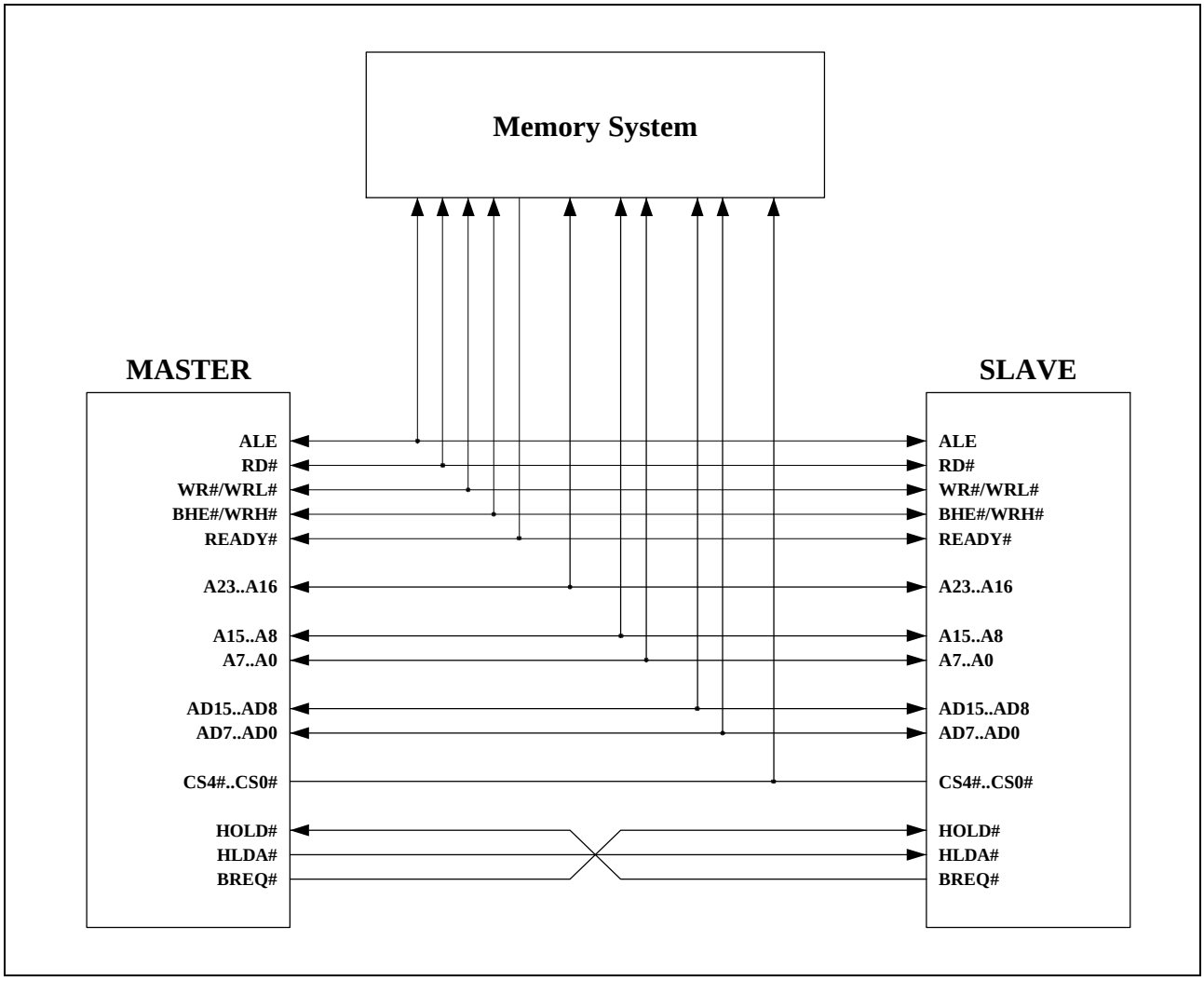

Figure 1:
Master/Slave Configuration with Shared Memory System

The master is defined as the one who is normally operating on the external bus, performing instruction fetches and data read/writes from/to the external memory. The slave is defined to normally execute out of internal resources, such as its internal ROM or RAM. Bus arbitration is required when the slave needs to also access portions of the external memory or XBUS peripherals of the master.

For this purpose, the external busses of the master and the slave are directly connected together. However, it must be always assured that at one time only one of them, either the master or the slave, controls all the external bus signals, while the other one puts ist bus pins into an high-impedance state. This arbitration of the external bus is controlled by the low-level active pins HOLD# (hold request), HLDA# (hold acknowledge), and BREQ# (bus request) of the two devices.

Now the definition and function of the bus arbitration signals is different in master and slave mode. The following table describes these differences.

Pin	Direction	Function in Master Mode
HOLD#	Input	While HOLD# is high, the master is operating in normal mode. A high-to-low transition issues a hold request from the master. The master backs off the bus, activates HLDA# and goes into hold mode. A low-to-high transitions issues the exit from hold mode. The master deactivates HLDA#, takes over the bus and enters normal operation again.
HLDA#	Output	High during normal operation. When the master enters hold mode, it sets HLDA# to low after releasing the bus. On exit of hold mode, the master first sets HLDA# to high and then goes onto the bus again.
BREQ#	Output	High during normal operation. The master activates BREQ# by setting it to low earliest one TCL after activating HLDA# if it has to perform an external bus access or XBUS access. If the master has regained the bus, BREQ# is set to high one TCL after deactivation of HLDA#.

Pin	Direction	Function in Slave Mode
HOLD#	Input	While both HOLD# and HLDA# are high, the slave is in hold mode, the bus interface is tristated. When the slave is released out of hold mode (HLDA# = 0) and has completely taken control over the external bus, a low level at this pins requests the slave to go into hold mode again. However, in any case the slave will perform at least one external bus cycle before going into hold mode again.
HLDA#	Input	A high-to-low transition at this pin releases the slave from hold mode.
BREQ#	Output	This signal is high as long as the slave operates from internal memory. When it detects that an external access is required, it sets BREQ# to low and waits for signal HLDA# to become low. BREQ# will go back to high when the slave has backed off the bus after it was requested to go into hold mode.

2 Initialization

Figure 2 shows the correct connection of the bus arbitration signals between the master and the slave. In order to provide correct levels during initialization of the master and the slave, two external pullup devices are required. One is connected to the master's HOLD# input, the other to the slave's HLDA# input.

Note: For compatibility reasons with existing applications, these pullups can not be integrated into the chip.

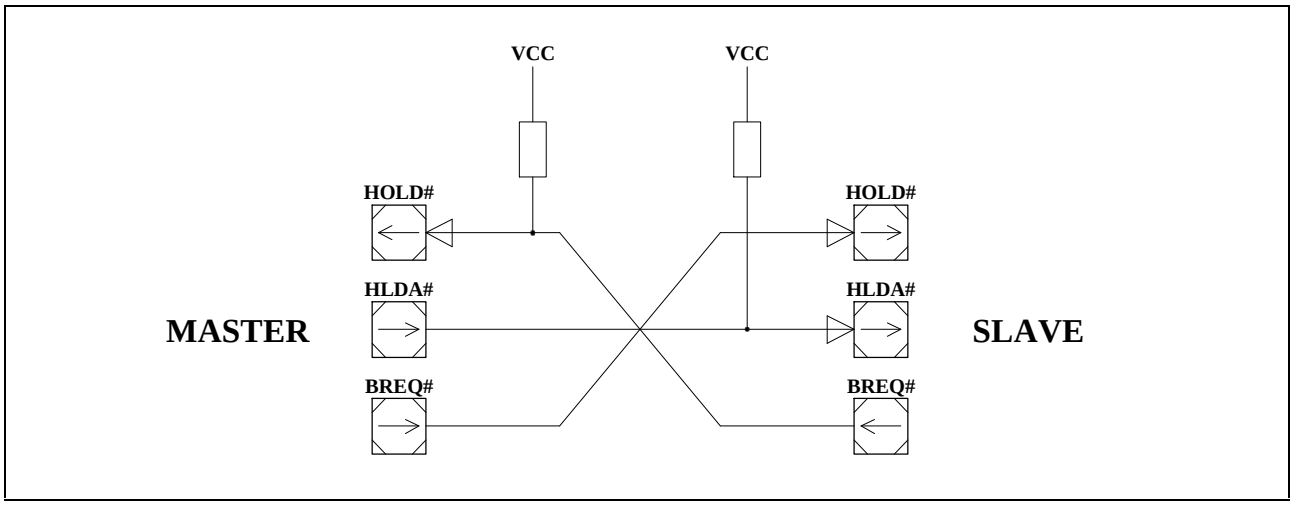

Figure 2:
Connection of the Master and Slave Bus Arbitration Signals

2.1 Master Initialization

After reset, the master is normally starting execution out of external memory. During the initialization, the HLDEN bit in register PSW must be set. Since the HOLD# pin is held high through the external pullup, no hold requests can occur even when the slave is not initialized yet.

If the master also wants to share its internal XBUS peripherals with the slave, it has to set the XPER_SHARE bit in register SYSCON (before the EINIT instruction).

Note that the HLDEN bit of the master can be reset during normal operation to force the master to ignore hold requests from the slave until HLDEN is set again. However, the pins HOLD#, HLDA# and BREQ# are still reserved for the bus arbitration and can not be used for general purpose I/O ! Only through an reset these pins can be switched to normal I/O again. This is intended to have the option to disable certain instruction sequences against interruption through hold requests.

2.2 Slave Initialization

The slave must start execution out of internal memory after reset. During initialization, the slave mode must first be selected. this is done by setting the direction bit of pin BREQ# to output (DP6.7 = 1). This enables the slave mode of the bus arbitration signals. After this, the HLDEN bit in register PSW must be set.

If the slave also wants to share its internal XBUS peripherals (if existing) with the master, it has to set the XPER_SHARE bit in register SYSCON (before the EINIT instruction).

Note: After setting the slave's HLDEN bit, the BREQ# output of the slave might be activated to low for a period of 2TCL. If the master does not recognize this hold request (it depends on the master's transition detection time slot, whether this short pulse is detected), this pulse has no effect. If the master recognizes this pulse, it might go into hold mode for one cycle. The exact timing in this case will be defined later.

Note: The effect of resetting bit HLDEN in slave mode (whether the slave is in hold or in normal mode) will be defined later. It is recommended to not reset the slave's HLDEN bit after initialization.

3 Operation of the master/slave bus arbitration

Figure 3 shows the sequence of the bus arbitration signals in a master/slave system. The startup condition is that the master is in normal mode and operating on the external bus, while the slave is in hold mode, operating from internal memory; the slave's bus interface is tristated. The marked time points in the diagram are explained in detail in the following.

1) The slave detects that it has to perform an external bus access. It activates BREQ# to low, which issues a hold request from the master.

2) The master activates HLDA# after releasing the bus. This initiates the slave's exit from hold sequence.

3a) When the master detects that it also has to perform external bus accesses (or accesses to its internal XBUS peripherals), it activates BREQ# to low. The earliest time for the master to activate BREQ# is one TCL after the activation of the master's HLDA# signal. However, the slave will ignore this signal until it has completely taken over control of the external bus. In this way, it is assured that the slave will at least perform one complete external bus access.

3b) If the master can operate from internal memory while it is in hold mode, it leaves the BREQ# signal high until it detects that an external bus access (or XBUS access) has to be performed. The slave therefore can stay on the bus as long as the master does not request the bus again.

4) When the master has requested the bus again through activation of its BREQ# signal, the slave will complete the current access and go into hold mode again. After completely tristateing its bus interface, the slave deactivates its BREQ# signal, thus releasing the master out of hold mode.

5) The master has terminated its hold mode and deactivates its HLDA# signal again. Now the master again controls the external bus again.

6) The master deactivates its BREQ# signal again one TCL after deactivation of HLDA#. From now on (and not earlier), the slave can generate a new hold request from the master. With this procedure it is assured that the master can perform at least one complete bus cycle before requested to go into hold mode again by the slave.

Also shown in Figure 3 is the sequence of the bus control between the master and the slave. The state of the bus signals between the master/slave bus control will be later defined in detail.

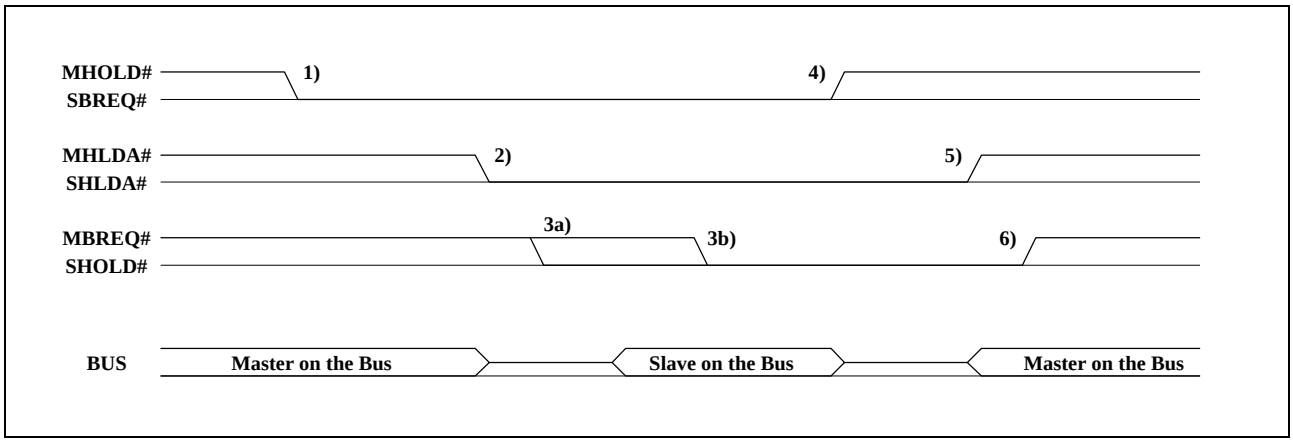

Figure 3:
Bus Arbitration Sequence