Data Sheet For 8259A Interrupt Control Unit

DOCUMENT REVISION HISTORY

Revision	Date	Change Description	Author
1.0	24th Dec'11	Initial Version	RS
REL 1.0	16th Aug'12	Removed implementation results	VC

Table of Contents

List Of Figures

List Of Tables

1 Introduction

1.1 Purpose

This document describes the Technical Specification 8259A Interrupt control unit. It includes the overall features, detailed description, I/O specifications and resource utilization summary for the 8259 Interrupt control unit.

1.2 Features

The following lists the main features of the 8259A Interrupt Controller:

- Eight interrupt request input per chip (INTP0 to INTP7).
- Up to 64 interrupt request inputs per system (Extended mode).
- Edge or level triggered interrupt request inputs.
- Individually maskable interrupt requests.
- Programmable interrupt request priority orders.
- Polling operation capability.
- Extended mode with cascade connection of external interrupts

1.3 Acronyms and Abbreviations

Table 1: Acronyms & Abbreviations

Term	Meaning
IRR	Interrupt Request Register
IMR	Interrupt Mask Register
ISR	Interrupt Service Register
IWn	Initialization Word n (n=1,2,3,4)
PFCW	Priority and Finish control word
IMW	Interrupt Mask Word

2 Interrupt Controller

2.1 Block Diagram

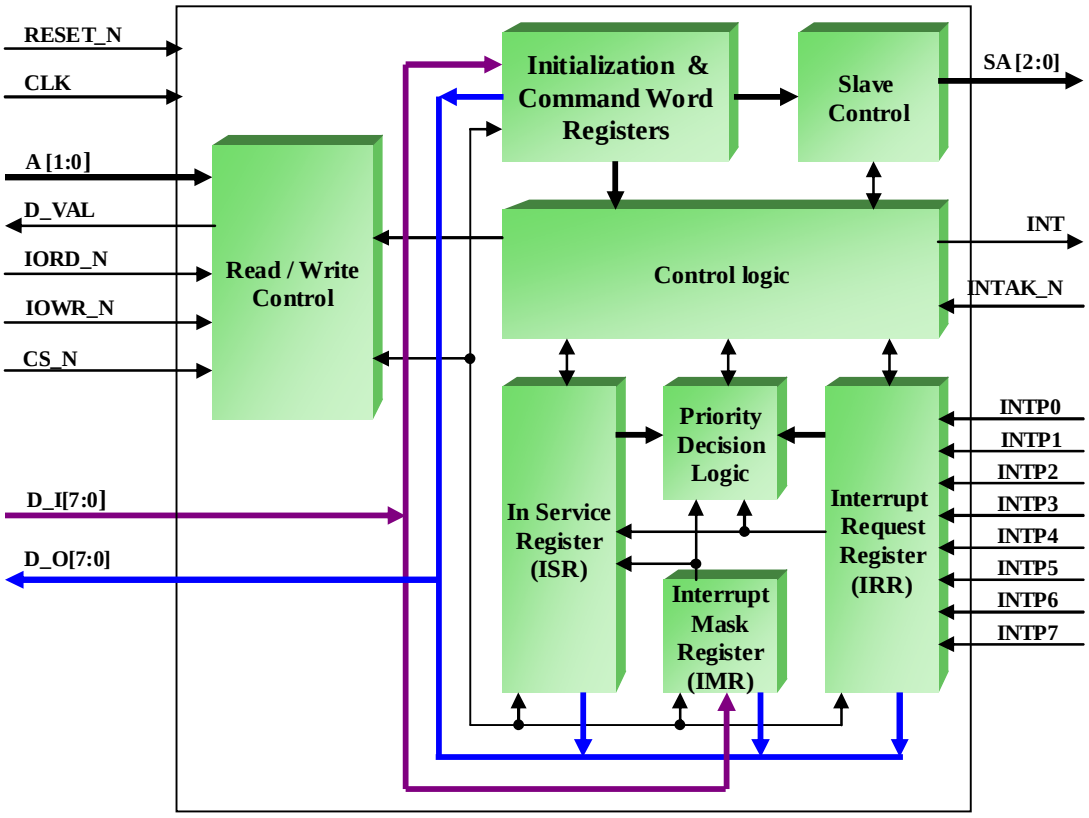

Figure 1: 8259A Interrupt Controller Block Diagram

2.2 Description

The 8259A Interrupt Controller can process eight interrupt request inputs, by allocating a priority level to the each request. It transfers the interrupt with highest priority to the CPU, along with interrupt address information. By cascading up to 64 interrupt requests can be processed. Interrupt routine address, interrupt request priority and masking are all under complete program control.

- **Read/Write Control:** The read/write control controls the CPU's reading and writing to and from the controller registers.
- **Initialisation and Command Word Registers:** These registers store the initialising words IW1-IW4 and command words PFCW (Priority and Finish Control Word), MCW (Mode Control Word) and IMW (Interrupt Mask Register). The CPU cannot read these registers.

- **Interrupt Mask Register [IMR]:** IMR stores the interrupt mask word (IMW) command word. Each bit masks an interrupts. If bit n of this register is 1, the interrupt request INTPn is masked and cannot be accepted by the controller. The CPU can read or write this register.
- **Interrupt Request Register [IRR]:** The Interrupt request register shows all interrupt levels are currently being requested. If bit n of the IRR is 1, INTPn is requested an interrupt. The CPU can read this register.
- **Interrupt Service Register [ISR]:** The ISR shows all interrupt levels currently in service. If bit n of this register is 1, the interrupt routine corresponding to INTPn is currently being executed. The CPU can read this register.
- **Slave Control:** Slave control is used for cascading purpose.
- **Control Logic:** The control logic receives and generates the signals that control the sequence of events in an interrupts.
- **Priority Decision Logic:** the priority decision logic determines which request from the IRR will be serviced next. The decision is made based upon the current interrupt mask, interrupt service status, mode status and current priority.

2.3 I/O Signal Description

Table 2: Interrupt Controller Signal Description

SIGNAL	I/O	WIDTH	DESCRIPTION
CLK	I	1	Processor Clock signal
RESET_N	I	1	Active low asynchronous reset signal
D_I [7:0]	I	8	Input DATA BUS.
D_O[7:0]	I	8	Output DATA BUS.
D_VAL	O	1	Output data valid signal.
IORD_N	I	1	Read Strobe.
IOWR_N	I	1	Write Strobe.
A [1:0]	I	1	Used with IORD_N, IOWR_N and CS_N to read or write to the Controller. A [1:0] = [A1, A0] if IOAG of SCTL Register is 1. A [1:0] = [A2, A1] if IOAG of SCTL Register is 0.
CS_N	I	1	Chip Select Input.
INTP [7:0]	I	8	8 Interrupt Request from the Peripherals.
INT	O	1	Interrupt Request Output from the Controller to the CPU (or Master).
INTAK_N	I	1	The Interrupt acknowledgement from the CPU.
SA [2 :0]	O	3	3 bit Slave Address bus in case of cascading.

3 Timing Waveforms

3.1 Interrupt Controller

3.1.1 Interrupt Controller Read Cycle

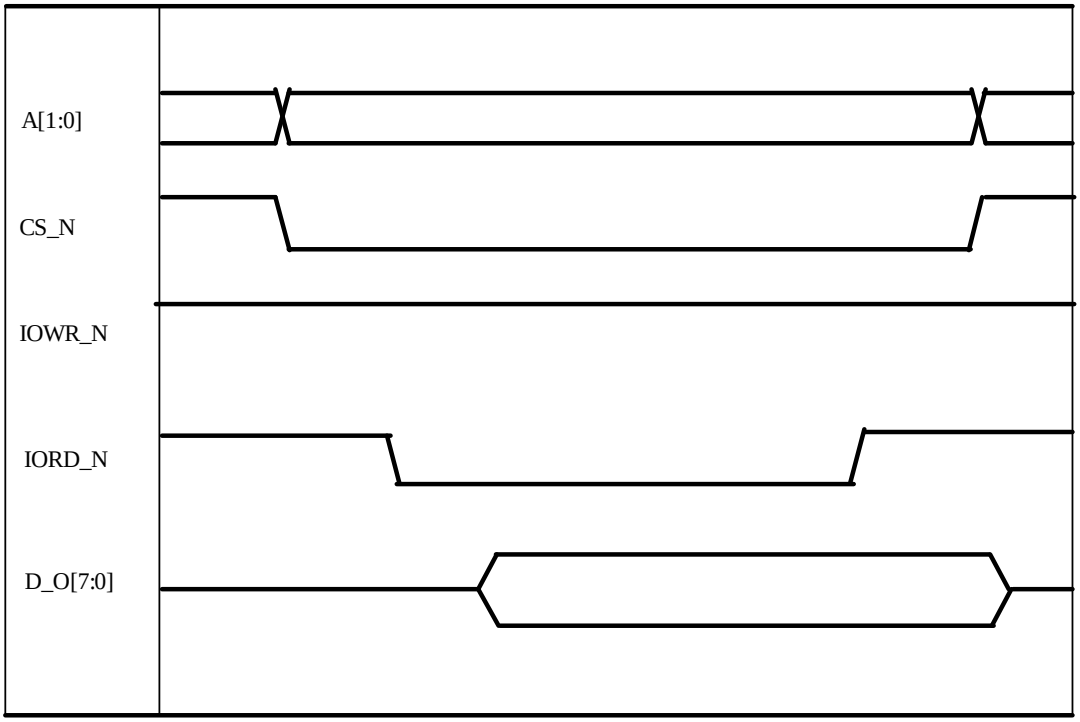

Figure 2: Interrupt Controller Read Cycle

3.1.2 Interrupt Controller Write Cycle

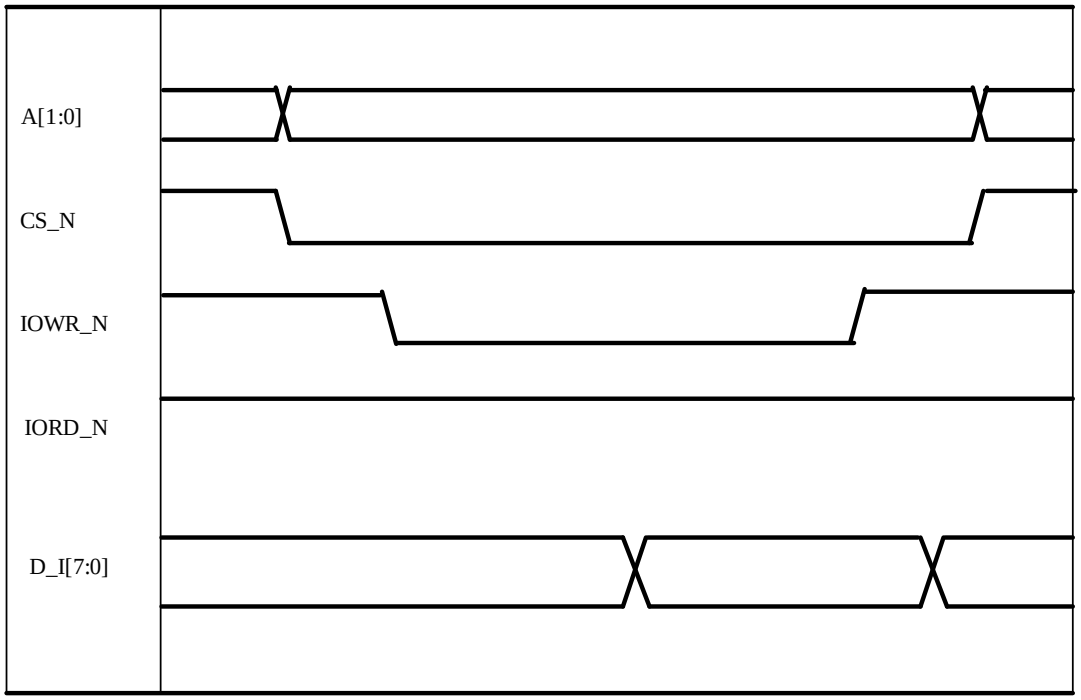

Figure 3: Interrupt Controller Write Cycle

3.1.3 Interrupt Acknowledgement Sequence for Vector Mode

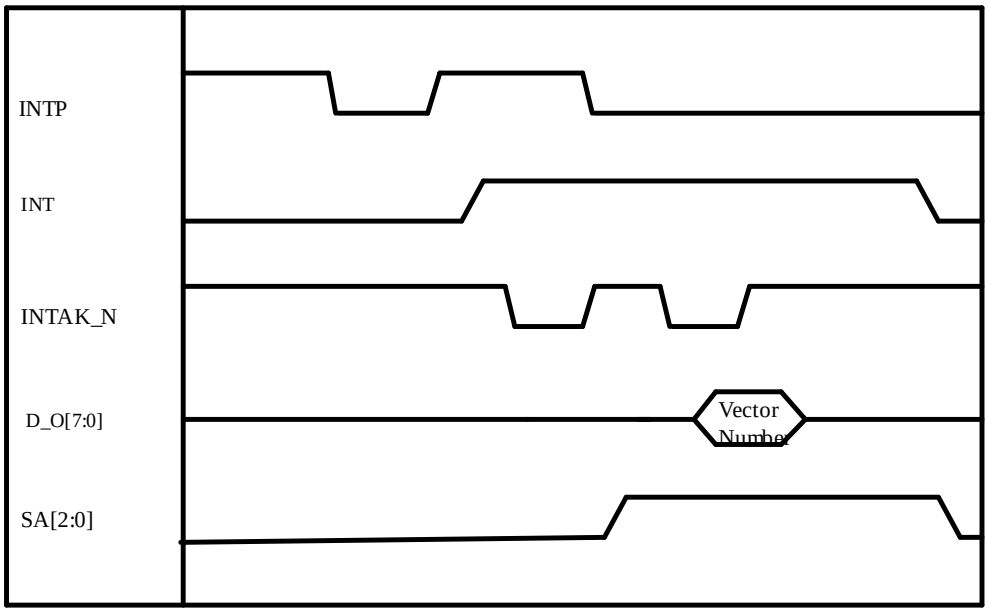

Figure 4: Interrupt Acknowledgement Sequence for Vector Mode