Z90230 Family of

Digital Television Controllers

User's Manual

PREFACE

0.1 PURPOSE

This user manual provides a comprehensive document that serves as a one-stop reference.

- Z90230 Family of Digital Television Controllers (DTCs)—Chapters 1, 2, 4, 6, and 7 contain information that directly relates to the Z90230-family components: General Description, Architectural Overview, Memory Registers, On-Screen Display, Input/Output Ports, and the Infrared Interface.
- Internal Microprocessor Overview—Chapter 3 contains information about the microcontroller-base functions used within the Z90230 family of products.
- I^2C Standard—Chapter 5 contains information about the implementation of the I^2C bus with the Z90230 products. Appendix A contains a copy of the I^2C Standard.
- Additional Reference—Appendices B and C contain reference information about Analog Peripherals and Support Tools.
- Quick Reference—Appendix D contains a quick reference of Memory Registers for experienced technical personnel. The Glossary provides an easy guide to acronyms and terminology.

In addition, the detailed Index combines with the Table of Contents, List of Figures, and List of Tables to make information easier to access. Essential information is at your fingertips, eliminating the need to cross-reference separate sources.

0.2 Z90230 FAMILY OF PRODUCTS

Z90230 represents a number of individual products. Please be aware that not all information within the manual applies to all products. Specific product applicability and exceptions may exist. Please check the Product Specification for the latest technical information on all supported devices.

0.3 NOTATION

The following conventions have been adopted for use throughout this manual:

- The notation 'addr (n)' is used to refer to bit 'n' of a given location. For example, bit 7 of the dst operand is referenced as :

  dst (7)

  Bits 4, 3, 2, 1, and 0 of the FADE_POS register are referenced as:

  FADE_POS (4,3,2,1,0)

- When the binary contents of a register are included in a text paragraph, the number appears as a series of 1s and 0s followed by B. For example:

  11001110B

- A register is described in a figure with the following format:

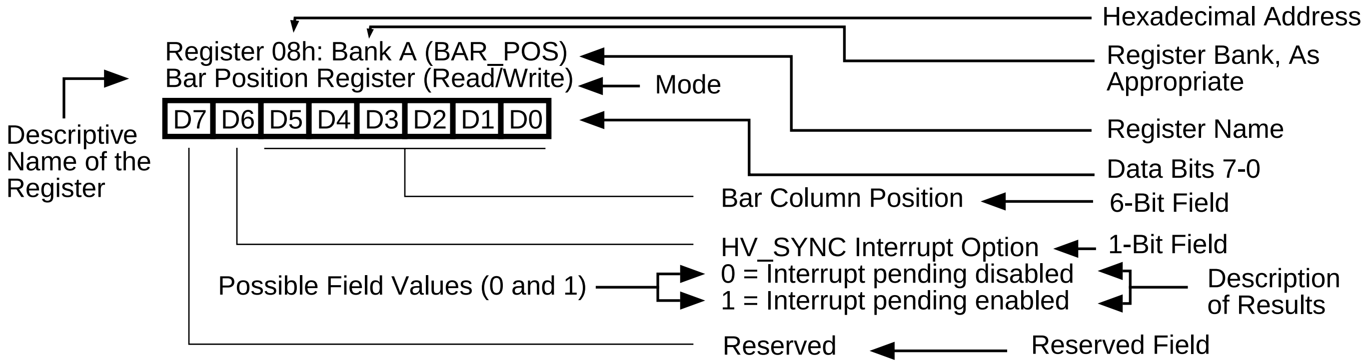


Figure 0-1. Example of Register Notation

Register Bits are numbered from right to left, 0 through 7. A letter may appear in the bit place to indicate the type of information stored in the bit. The following letters designate the bit type or value:

D	Data Bit
T	Timer Bit
U	Unknown Value
X	Place Holder
0	Binary Value 0
1	Binary Value 1

- The following codes appear within tables:

AI	Analog Input
I	Input
NC	Not Connected
O	Output
PWR	Power
R	Read
W	Write
%D	Data

0.4 I²C

The I²C bus is licensed by Zilog Inc. from Philips International BV. The terms of the license agreement require display of the following notice:

> Purchase of I²C components of Zilog Inc. or one of its sublicensed Associated Companies conveys a license under the Philips I²C Patent Rights to use these components in an I²C system, provided that the system conforms to the I²C standard specification as defined by Philips.

0.5 ABOUT THIS USER'S MANUAL

The following individuals have contributed to the preparation of this User's Manual: Laura Bayer, Dongsoo Kim, Bruno Kranzen, Steven Lau, Alexander Marquez, Alex Muratov, Donghyun Song, and Anatoliy Tsyrganovich.

Z90230 FAMILY OF DTCS USER'S MANUAL

TABLE OF CONTENTS

CHAPTER TITLE AND SUBSECTIONS PAGE

Chapter 1. Introduction
1.1. Features 1-1
1.2. General Description 1-1

Chapter 2. Architectural Overview
2.1. Introduction 2-1
2.2. Hardware 2-1
2.2.1. Pin Identification 2-1
2.2.2. Z90239 124-Pin PGA Ceramic Package Pin-Out Diagram 2-4
2.2.3. Z90239 Pin Assignment 2-4
2.2.4. Pin Descriptions 2-7
2.2.5. Core Customization 2-9
2.2.6. Block Diagram 2-10
2.3. Control Registers 2-11
2.3.1. 3-Bit ADC Data Register 2-11
2.3.2. 4-Bit ADC Data Register 2-12
2.3.3. Port 4 Pin-Out Selection Register 2-13
2.3.4. Expanded Register File 2-14
2.3.5. Stop-Mode Recovery Register 2-16
2.3.6. Watch-Dog Timer Mode Register 2-17
2.3.7. Timer Mode Register 2-17
2.3.8. Counter/Timer 1 Register 2-18
2.3.9. Prescaler 1 Register 2-18
2.3.10. Counter/Timer 0 Register 2-18
2.3.11. Prescaler 0 Register 2-19
2.3.12. Port 2 Mode Register 2-20
2.3.13. Port 2 Control Register 2-20
2.3.14. Interrupt Priority Register 2-21
2.3.15. Interrupt Request Register 2-22
2.3.16. Interrupt Mask Register 2-23
2.3.17. Flags Register 2-24
2.3.18. Register Pointer 2-24
2.3.19. Stack Pointer High 2-24

CHAPTER TITLE AND SUBSECTIONS PAGE

Chapter 2. Architectural Overview (Continued)

2.3. Control Registers (Continued)
2.3.20. Stack Pointer Low 2-25
2.3.21. Port 2 Data Register 2-25
2.4. Operating Characteristics 2-26
2.4.1. DC Characteristics 2-27
2.4.2. AC Characteristics 2-28

Chapter 3. Internal Microprocessor Overview

3.1. Address Space 3-1
3.2. Standard Register File 3-1
3.2.1. General-Purpose Registers 3-3
3.2.2. RAM Protect 3-3
3.2.3. Working Register Groups 3-3
3.2.4. Error Conditions 3-4
3.3. Expanded Register File 3-5
3.4. Control and Peripheral Registers 3-8
3.4.1. Standard Registers 3-8
3.4.2. Expanded Registers 3-8
3.5. Program Memory 3-10
3.6. Stacks 3-11
3.7. Oscillator Control 3-12
3.8. Oscillator Operation 3-12
3.8.1. Layout 3-13
3.8.2. Indications of an Unreliable Design 3-13
3.8.3. Circuit Board Design Rules 3-14
3.8.4. Crystals and Resonators 3-15
3.9. LC Oscillator 3-16
3.10. RESET—Watch-Dog Timer 3-16
3.11. Reset Pin, Internal POR Operation 3-17
3.12. Watch-Dog Timer 3-20
3.13. Power-On Reset 3-21
3.14. Counter/Timers 3-21
3.15. Prescalers and Counter/Timers 3-22
3.16. Counter/Timers Operation 3-24
3.16.1. Load and Enable Count Bits 3-24
3.16.2. Prescaler Operations 3-25
3.17. T_{IN} Mode 3-26
3.17.1. H_{SYNC} Clock Input Mode 3-27
3.18. Counter/Timer Reset Conditions 3-27
3.19. Interrupts 3-30
3.20. Interrupt Sources 3-31
3.20.1. External Interrupt Source 3-31
3.21. Interrupt Request Register Logic and Timing 3-32

CHAPTER TITLE AND SUBSECTIONS **PAGE**

Chapter 3. Internal Microprocessor Overview (Continued)

3.22. Interrupt Initialization 3-32
3.22.1. Interrupt Priority Register Initialization 3-33
3.22.2. Interrupt Mask Register Initialization 3-34
3.22.3. Interrupt Request Register Initialization 3-35
3.23. IRQ Software Interrupt Generation 3-37
3.24. Vectored Processing 3-38
3.24.1. Vectored Interrupt Cycle Timing 3-39
3.24.2. Nesting of Vectored Interrupts 3-40
3.25. Polled Processing 3-40
3.26. Interrupt Reset Conditions 3-41
3.27. Power-Down Halt-Mode Operation 3-41
3.28. Stop-Mode Operation 3-41
3.29. STOP-Mode Recovery Register 3-42
3.30. Addressing Modes 3-45
3.31. Register Addressing 3-45
3.32. Indirect Register Addressing 3-46
3.33. Indexed Addressing 3-49
3.34. Direct Addressing 3-50
3.35. Relative Addressing 3-51
3.36. Immediate Data Addressing 3-52
3.37. Instruction Set Functional Summary 3-52
3.38. Processor Flags 3-54
3.38.1. Carry Flag 3-55
3.38.2. Zero Flag 3-55
3.38.3. Sign Flag 3-55
3.38.4. Overflow Flag 3-55
3.38.5. Decimal-Adjust Flag 3-55
3.38.6. Half-Carry Flag 3-55
3.39. Condition Codes 3-56
3.40. Notation and Binary Encoding 3-57
3.40.1. Assembly Language Syntax 3-58

Chapter 4. On-Screen Display

4.1. Introduction 4-1
4.2. OSD Position 4-2
4.2.1. OSD Control Register 4-3
4.2.2. Vertical Position Register 4-4
4.2.3. Horizontal Position Register 4-4
4.2.4. Second Color Feature 4-5
4.2.5. Second Color Control Register 4-5
4.2.6. Second Color Register 4-6
4.2.7. Second Color Example 4-6
4.3. Mesh and Halftone Effect 4-7
4.3.1. Mesh Column Start Register 4-9
4.3.2. Mesh Column End Register 4-10
4.3.3. Mesh Row Enable Register 4-10

CHAPTER TITLE AND SUBSECTIONS **PAGE**

Chapter 4. On-Screen Display (Continued)

4.3. Mesh and Halftone Effect (Continued)
4.3.4. Mesh Control Register 4-11
4.3.5. Mesh Window Display Example 4-12
4.4. OSD Fade 4-15
4.5. Inter-Row Spacing 4-17
4.6. Character Generation 4-18
4.6.1. Character Cell Resolution 4-18
4.6.2. Character Size and Smoothing Effect 4-20
4.6.3. Fringing Effect 4-21
4.7. Display Attribute Control 4-22
4.7.1. Display Attribute Register 4-22
4.7.2. Video Refresh RAM Access 4-23
4.7.3. Color Table and Color Index Register 4-25
4.7.4. Row Attribute Register 4-27
4.8. HV Interrupt Processing 4-27
4.8.1. HV Interrupt Status Register 4-28
4.8.2. H_{SYNC} and V_{SYNC} Requirements 4-29
4.9. Dot Clock Oscillator 4-30
4.9.1. Layout 4-30

Chapter 5. I^2C Interface

5.1. I^2C-Bus Concepts 5-1
5.2. Data Validity 5-1
5.3. START and STOP Conditions 5-1
5.4. Data Transfer 5-1
5.5. Byte Format 5-2
5.6. Acknowledge 5-3
5.7. Z90230 Family I^2C Master Interface 5-3
5.7.1. Master I^2C Control Register 5-5
5.8. Software Control of the I^2C Interface 5-6

Chapter 6. Input/Output Ports

6.1. Input/Output Ports 6-1
6.1.1. Port Configuration Register 6-1
6.1.2. Port 2 Mode Register 6-2
6.1.3. Port 2 Data Register 6-3
6.1.4. Port 4 Pin-Out Selection Register 6-3
6.1.5. Port 4 Data Register 6-5
6.1.6. Port 4 Direction Control Register 6-6
6.1.7. Port 5 - PWM Mode Register 6-7
6.1.8. Port 5 Data Register 6-8
6.1.9. Port 5 Direction Control Register 6-9
6.1.10. Port 6 Data Register 6-10
6.1.11. Port 6 Direction Control Register 6-11

CHAPTER TITLE AND SUBSECTIONS PAGE

Chapter 7. Infrared Interface

7.1. Infrared Interface 7-1
7.1.1. Timer Control Register 0 7-2
7.1.2. Timer Control Register 1 7-3
7.1.3. IR Capture Register 0 7-4
7.1.4. IR Capture Register 1 7-4
7.1.5. IR Decoding 7-5

Chapter 8. Pulse Width Modulators

8.1. Pulse Width Modulators 8-1
8.1.1. PWM Mode Register 8-1
8.1.2. Port 4 Pin-Out Selection Register 8-2
8.1.3. PWM1 through PWM11 8-3
8.1.4. Digital/Analog Conversion via PWM 8-7

Appendix A. Philips I^{2}C Specification

A.1. Philips I^{2}C Specification A-1

Appendix B. Analog Peripherals

B.1. Analog-to-Digital Converter B-1
B.1.1. 3-Bit ADC Data Register B-2
B.1.2. 4-Bit ADC Data Register B-2
B.1.3. ADC Block Diagram B-3

Appendix C. Support Products

C.1. Z90230 Family Support Products C-1
C.1.1. ICEBOX Family In-Circuit Emulators C-1
C.1.2. Z90219 Emulator (Z9021901ZEM) C-1
C.1.3. Z90219 Emulation Module (Z9020900TSC) C-1
C.1.4. Z89332 Evaluation Board (Z8933200ZCO) C-1
C.1.5. ICEBOX/HP Logic Analyzer Adapter Board (Z89C0000ZHP) C-1
C.1.6. Zilog Macro Cross Assembler (ZMASM0W0ZAS) C-2
C.1.7. ZMASM Supported Cores/Devices C-2

Appendix D. Registers

D.1. Registers D-1

Appendix E. EMI/Noise Reduction

E.1. EMI/Noise Reduction Through PCB Design E-1

Appendix F. Sales Offices F-1

Appendix G. Literature Guide G-1

Z90230 FAMILY OF DTCS USER'S MANUAL

LIST OF FIGURES

NUMBER AND TITLE PAGE

Chapter 1. Introduction
Figure 1-1. Z90230 DTC System Application 1-2

Chapter 2. Architectural Overview
Figure 2-1. Z90231 and Z90233 Pin Identification 2-2
Figure 2-2. Z90239 124-Pin PGA Ceramic Package Pin-Out Diagram 2-4
Figure 2-3. Block Diagram 2-10
Figure 2-4. 3-Bit ADC Data Register 2-11
Figure 2-5. 4-Bit ADC Data Register 2-12
Figure 2-6. Port 4 Pin-Out Selection Register 2-13
Figure 2-7. Register and Expanded Register File Map 2-14
Figure 2-8. Expanded Register File 2-15
Figure 2-9. Stop-Mode Recovery Register 2-16
Figure 2-10. Watch-Dog Timer Mode Register 2-17
Figure 2-11. Timer Mode Register 2-17
Figure 2-12. CounterTimer1 Register 2-18
Figure 2-13. Prescaler 1 Register 2-18
Figure 2-14. Counter/Timer 0 Register 2-18
Figure 2-15. Prescaler 0 Register 2-19
Figure 2-16. Port 2 Mode Register 2-20
Figure 2-17. Port 2 Control Register 2-20
Figure 2-18. Interrupt Priority Register 2-21
Figure 2-19. Interrupt Request Register 2-22
Figure 2-20. Interrupt Mask Register 2-23
Figure 2-22. Register Pointer 2-24
Figure 2-23. Stack Pointer High Register 2-24
Figure 2-21. Flags Register 2-24
Figure 2-24. Stack Pointer Low Register 2-25
Figure 2-25. Port 2 Data Register 2-25
Figure 2-26. AC Characteristics 2-29

NUMBER AND TITLE PAGE

Chapter 3. Internal Microprocessor Overview

Figure 3-1. 16-Bit Register Addressing 3-2
Figure 3-2. Accessing Individual Bits (Example) 3-2
Figure 3-3. Working Register Addressing Examples 3-3
Figure 3-4. Register Pointer 3-4
Figure 3-5. Register and Expanded Register File Map 3-6
Figure 3-6. Register Pointer (FDh) Example 3-7
Figure 3-7. Program Memory Map 3-10
Figure 3-8. Stack Pointer 3-11
Figure 3-9. Stack Operations 3-11
Figure 3-10. Port Configuration Register 3-12
Figure 3-11. Pierce Oscillator with Internal Feedback Circuit 3-13
Figure 3-12. Circuit Board Design Rules 3-14
Figure 3-13. Crystal/Ceramic Resonator Oscillator 3-15
Figure 3-14. LC Clock 3-15
Figure 3-15. External Clock 3-15
Figure 3-16. Capacitance Calculation 3-16
Figure 3-17. Reset Timing 3-18
Figure 3-18. External Power-On Reset Circuit Example 3-18
Figure 3-19. Microprocessor Reset with Reset Pin, WDT, SMR, and POR (Example) 3-19
Figure 3-20. Watch-Dog Timer Mode Register (Write-Only) Example 3-20
Figure 3-21. Counter/Timers Block Diagram 3-22
Figure 3-22. Counter/Timers Register Map 3-23
Figure 3-23. Prescaler 1 Register 3-23
Figure 3-24. Prescaler 0 Register 3-24
Figure 3-25. Counter/Timer 0 and 1 Registers 3-24
Figure 3-26. Timer Mode Register 3-25
Figure 3-27. Starting The Count 3-25
Figure 3-28. Counting Modes 3-25
Figure 3-29. Timer Mode Register (T_{IN} Operation) 3-27
Figure 3-30. Prescaler 1 Register (T_{IN} Operation) 3-27
Figure 3-31. H_{SYNC} Clock Input Mode 3-27
Figure 3-32. Counter/Timer 1 Register After Reset 3-28
Figure 3-33. Counter/Timer 0 Register After Reset 3-28
Figure 3-34. Prescaler 1 Register After Reset 3-28
Figure 3-35. Prescaler 0 Register After Reset 3-29
Figure 3-36. Timer Mode Register After Reset 3-29
Figure 3-37. Interrupt Control Registers 3-30
Figure 3-38. Interrupt Block Diagram 3-30
Figure 3-39. Interrupt Sources IRQ0-IRQ2 Block Diagram 3-31
Figure 3-40. IRQ Register Logic 3-32
Figure 3-41. Interrupt Request Timing 3-32
Figure 3-42. Interrupt Priority Register 3-33
Figure 3-43. Interrupt Mask Register 3-35
Figure 3-44. Interrupt Request Register 3-36
Figure 3-45. IRQ Reset Functional Logic Diagram 3-37
Figure 3-46. Effects of an Interrupt on the Stack 3-38

NUMBER AND TITLE PAGE

Chapter 3. Internal Microprocessor Overview (Continued)

Figure 3-47. Interrupt Vectors in Memory 3-39
Figure 3-48. Interrupt Acknowledge Timing 3-40
Figure 3-49. Stop-Mode Recovery Register 3-43
Figure 3-50. Stop-Mode Recovery Source/Level Select 3-44
Figure 3-51. 8-Bit Register Addressing 3-45
Figure 3-52. 4-Bit Register Addressing 3-46
Figure 3-53. 8-Bit Indirect Register Addressing 3-47
Figure 3-54. 4-Bit Indirect Register Addressing 3-48
Figure 3-55. Indexed Register Addressing 3-49
Figure 3-56. Direct Addressing 3-50
Figure 3-57. Relative Addressing 3-51
Figure 3-58. Immediate Data Addressing 3-52
Figure 3-59. Flag Register 3-54

Chapter 4. On-Screen Display

Figure 4-1. OSD Format 4-2
Figure 4-2. OSD Control Register 4-3
Figure 4-3. Positive and Negative Sync Signals 4-3
Figure 4-4. Vertical Position Register 4-4
Figure 4-5. Vertical Position Example 4-4
Figure 4-6. Horizontal Position Register 4-4
Figure 4-7. Horizontal Position Example 4-5
Figure 4-8. Second Color Control Register 4-5
Figure 4-9. Second Color Register 4-6
Figure 4-10. Second Color Example 4-6
Figure 4-11. Second Color Example Registers 4-7
Figure 4-12. Mesh (Example) 4-8
Figure 4-13. Mesh On 4-9
Figure 4-14. Mesh Column Start Register 4-9
Figure 4-15. Mesh Column End Register 4-10
Figure 4-16. Mesh Row Enable Register 4-10
Figure 4-17. Mesh Control Register 4-11
Figure 4-18. Mesh Window Display Registers for Row 0 (Example) 4-13
Figure 4-19. Mesh Window Display Registers for Row 1-6 (Example) 4-14
Figure 4-20. Mesh Window Display Registers for Row 7 (Example) 4-15
Figure 4-21. Video Fade (Example) 4-16
Figure 4-22. Fade Position Register 1 4-17
Figure 4-23. Fade Position Register 2 4-17
Figure 4-24. Row Space Register 4-17
Figure 4-25. Character Pixel Map in CGROM (Example) 4-19
Figure 4-26. Icon Display 4-20
Figure 4-27. Smoothing 4-21
Figure 4-28. Display Attribute Register 4-22
Figure 4-29. VRAM Address Map 4-23
Figure 4-30. Color Palette Selection Bits Update 4-24
Figure 4-31. Color Index Register 4-25

NUMBER AND TITLE PAGE

Chapter 4. On-Screen Display (Continued)

Figure 4-32. Color Palette 0 4-25
Figure 4-33. Color Palette 1 4-26
Figure 4-34. Color Palette 2 4-26
Figure 4-35. Color Palette 3 4-26
Figure 4-36. Color Palette 4 4-26
Figure 4-37. Color Palette 5 4-26
Figure 4-39. Row Attribute Register 4-27
Figure 4-38. Color Palette 6 4-27
Figure 4-40. HV Interrupt Status Register 4-28
Figure 4-41. H_{SYNC} and V_{SYNC} Specification 4-29
Figure 4-42. Dot Clock Oscillator 4-30
Figure 4-43. Oscillation Frequency 4-30
Figure 4-44. Simple Series Capacitance 4-30

Chapter 5. I²C Interface

Figure 5-1. Data Transfer 5-2
Figure 5-2. Bidirectional Port Pin Pad Multiplexed with I^2C Port 5-4
Figure 5-3. Master I^2C Control Register 5-5
Figure 5-4. Master I^2C Command Register 5-6
Figure 5-5. Master I^2C Data Register 5-6
Figure 5-6. Data Frame Write Flowchart 5-8
Figure 5-7. Data Frame Read Flowchart 5-9

Chapter 6. Input/Output Ports

Figure 6-1. Port Configuration Register 6-1
Figure 6-2. Port 2 Mode Register 6-2
Figure 6-3. Port 2 Data Register 6-3
Figure 6-4. Port 4 Pin-Out Selection Register 6-4
Figure 6-5. Port 4 Data Register 6-5
Figure 6-6. Port 4 Direction Control Register 6-6
Figure 6-7. PWM Mode Register 6-7
Figure 6-8. Port 5 Data Register 6-8
Figure 6-9. Port 5 Direction Control Register 6-9
Figure 6-10. Port 6 Data Register 6-10
Figure 6-11. Port 6 Direction Control Register 6-11

Chapter 7. Infrared Interface

Figure 7-1. Timer Control Register 0 7-2
Figure 7-2. Timer Control Register 1 7-3
Figure 7-3. IR Capture Register 0.... 7-4
Figure 7-4. IR Capture Register 1 7-4
Figure 7-5. IR Decoding Flowchart Example 7-5

NUMBER AND TITLE PAGE

Chapter 8. Pulse Width Modulators

Figure 8-1. PWM Mode Register 8-2
Figure 8-2. Port 4 Pin-Out Selection Register 8-3
Figure 8-3. Pulse Width Modulator Timing Diagram, 6 Bit 8-4
Figure 8-4. Pulse Width Modulator Timing Diagram, 14 Bit 8-5
Figure 8-5. PWM1 through PWM10 Registers 8-6
Figure 8-6. PWM11 Register 8-7
Figure 8-7. Analog Signals Generated from PWM Signals 8-8

Appendix B. Analog Peripherals

Figure B-1. 3-Bit ADC Data Register B-2
Figure B-2. 4-Bit ADC Data Register B-2
Figure B-3. ADC Block Diagram B-3

Appendix D. Registers

Figure D-1. Expanded Register File D-2
Figure D-2. Register and Expanded Register File Map D-3
Figure D-3. OSD Control Register D-4
Figure D-4. Vertical Position Register D-4
Figure D-5. Horizontal Position Register D-4
Figure D-6. Display Attribute Register D-5
Figure D-7. Row Space Register D-5
Figure D-8. Fade Position Register 1 .... D-6
Figure D-9. Fade Position Register 2 .... D-6
Figure D-10. Second Color Control Register D-6
Figure D-11. Second Color Register D-6
Figure D-12. Color Palette 0 .... D-7
Figure D-13. Color Palette 1 .... D-7
Figure D-14. Color Palette 2 .... D-7
Figure D-15. Color Palette 3 .... D-7
Figure D-16. Color Palette 4 .... D-8
Figure D-17. Color Palette 5 .... D-8
Figure D-18. Color Palette 6 .... D-8
Figure D-19. PWM11 Register D-8
Figure D-20. PWM1 through PWM10 Registers D-9
Figure D-21. Row Attribute Register D-10
Figure D-22. Port 5 Data Register D-10
Figure D-23. PWM Mode Register D-11
Figure D-24. Port 5 Direction Control Register D-12
Figure D-25. 3-Bit ADC Data Register D-12
Figure D-26. Timer Control Register 0 .... D-13
Figure D-27. Timer Control Register 1 .... D-13
Figure D-28. IR Capture Register 0 .... D-14
Figure D-29. IR Capture Register 1 .... D-14
Figure D-30. Port 4 Data Register D-14
Figure D-31. Port 4 Direction Control Register D-15
Figure D-32. HV Interrupt Status Register D-16

NUMBER AND TITLE PAGE

Appendix D. Registers (Continued)

Figure D-33. Port 4 Pin-Out Selection Register D-16
Figure D-34. Color Index Register D-17
Figure D-35. Master I^2C Data Register D-17
Figure D-36. Master I^2C Command Register D-17
Figure D-37. Master I^2C Control Register D-18
Figure D-38. Port Configuration Register D-19
Figure D-39. 4-Bit ADC Data Register D-19
Figure D-40. Port 6 Direction Control Register D-20
Figure D-41. Port 6 Data Register D-21
Figure D-42. Mesh Column Start Register D-21
Figure D-43. Mesh Column End Register D-21
Figure D-44. Mesh Row Enable Register D-22
Figure D-45. Mesh Control Register D-23
Figure D-46. Stop-Mode Recovery Register D-24
Figure D-47. Watch-Dog Timer Mode Register D-24
Figure D-48. Stack Pointer Low Register D-25
Figure D-49. Stack Pointer High Register D-25
Figure D-50. Register Pointer D-25
Figure D-51. Flag Register D-25
Figure D-52. Interrupt Mask Register D-26
Figure D-53. Interrupt Request Register D-26
Figure D-54. Interrupt Priority Register D-27
Figure D-55. Port 2 Control Register D-27
Figure D-56. Port 2 Mode Register D-28
Figure D-57. Prescaler 0 Register D-28
Figure D-58. Counter/Timer 0 Register D-29
Figure D-59. Prescaler 1 Register D-29
Figure D-60. Counter/Timer 1 Register D-29
Figure D-61. Timer Mode Register D-30
Figure D-62. Port 2 Data Register D-30

Appendix E. EMI/Noise Reduction

Figure E-1. Application Circuit E-1

Z90230 FAMILY OF DTCS USER'S MANUAL

LIST OF TABLES

NUMBER AND TITLE **PAGE**

Chapter 1. Introduction

Table 1-1. Z90200-Family Product Summary 1-3

Chapter 2. Architectural Overview

Table 2-1. Z90230-Family OTP and Production Pin Assignment 2-3
Table 2-2. Z90239 Pin Assignments 2-4
Table 2-3. IRQ Function Summary 2-22
Table 2-4. IRQ Register Configuration 2-23
Table 2-5. Operational Limits 2-26
Table 2-6. DC Characteristics 2-27
Table 2-7. AC Characteristics 2-28

Chapter 3. Internal Microprocessor Overview

Table 3-1. Working Register Groups 3-2
Table 3-2. Expanded Register File Bank A 3-8
Table 3-3. Expanded Register File Bank B 3-9
Table 3-4. Expanded Register File Bank C 3-9
Table 3-5. Expanded Register File Bank F 3-9
Table 3-6. Sample Control and Peripheral Register Reset Values 3-17
Table 3-7. Expanded Register File Bank 0 Reset Values at Reset 3-19
Table 3-8. Time-Out Period of the WDT 3-20
Table 3-9. Interrupt Types, Sources, and Vectors 3-31
Table 3-10. Interrupt Priority 3-34
Table 3-11. Interrupt Group Priority 3-34
Table 3-12. IRQ Function Summary 3-36
Table 3-13. IRQ Register Configuration 3-37
Table 3-14. Stop-Mode Recovery Source 3-43
Table 3-15. Load Instructions 3-53
Table 3-16. Arithmetic Instructions 3-53
Table 3-17. Logical Instructions 3-53
Table 3-18. Program Control Instructions 3-53
Table 3-19. Bit Manipulation Instructions 3-53
Table 3-20. Block Transfer Instructions 3-53

NUMBER AND TITLE PAGE

Chapter 3. Internal Microprocessor Overview (Continued)

Table 3-21. Rotate and Shift Instructions 3-53
Table 3-22. CPU Control Instructions 3-54
Table 3-23. Flag Definitions 3-56
Table 3-24. Flag Settings Definitions 3-56
Table 3-25. Condition Codes 3-56
Table 3-26. Notational Shorthand 3-57
Table 3-27. Additional Notation 3-58

Chapter 4. On-Screen Display

Table 4-1. BGR Mesh Colors 4-12
Table 4-2. RGB Colors 4-21
Table 4-3. Color Palette Selection Bits 4-24

Chapter 5. I^{2}C Interface

Table 5-1. Master I^{2}C Bus Interface Commands 5-7

Chapter 7. Infrared Interface

Table 7-1. IR Interrupt Captured Values 7-2
Table 7-2. IR Capture Timer Speed Setting 7-3

Chapter 8. Pulse Width Modulators

Table 8-1. Expanded Register File Bank B 8-1
Table 8-2. Pulse Width Modulator Pin Functional Description Example 8-1

CHAPTER 1
INTRODUCTION

1.1 FEATURES

The Z90230 Family of Digital Television Controllers (DTCs) features a variety of RAM and ROM options; together with a host of advanced On-Screen Display (OSD) features to support high-end graphics. The display resolution is particularly suitable for Asian languages.

Advanced features include:

- New Color Palette System
- Flexible Inter-Row Spacing
- Higher Character Cell Resolution
- Halftone Effect
- Window-Based Background Mesh Effect
- Dedicated Infrared Interface
- On-Chip Analog-to-Digital Conversion
- VRAM and Increased System ROM
- Hardware Master-Mode I^2C Interface

The memory efficient core in combination with these advanced features makes the Z90230 DTC family an ideal choice in the PAL, SECAM, and NTSC markets.

1.2 GENERAL DESCRIPTION

The Z90200 DTC family consists of three basic device types, Z90200, Z90220, and Z90230. The Z90200 family supports the I^2C communication standard via software.

The Z90220 family supports closed-caption decoding (CCD), and is currently under development. The Z90230 family supplies a standard I^2C communication port, half-tone OSD circuitry, and programmable two-pin I/O assignment.

Figure 1-1 illustrates how the Z90230 DTC can be used as an application-specific controller designed to provide complete audio and video control of television receivers and video recorders, and advanced on-screen display facilities.

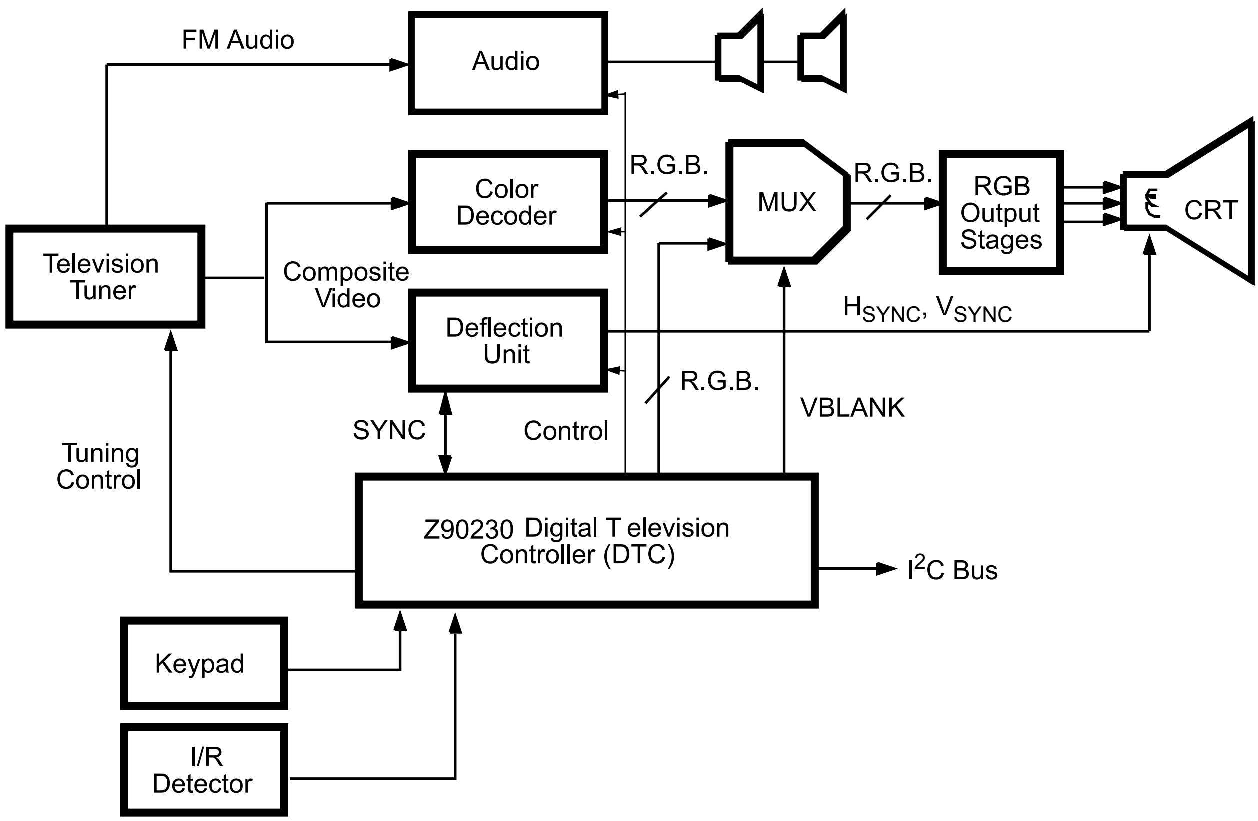


Figure 1-1. Z90230 DTC System Application

The Z90200 family takes full advantage of the Z8's expanded register file space to offer greater flexibility in creating a user-friendly on-screen display.

Three basic addressing spaces are available: program memory, Video RAM (VRAM), and the register file. The register file is composed of 236 bytes of general-purpose registers, 16 control and status registers, 1 I/O port register, and 2 reserved registers.

The on-screen display control circuits support 10 rows by 24 columns (10x24) of characters. The character color is specified per character. There are eight foreground colors and eight background colors. When foreground and background colors are the same, the background is transparent. An analog bar line can be displayed when settings are defined for Row, Second Color, and Character Set. The bar is used to display volume control, signal levels and tuning.

The OSD is capable of displaying 2 character sizes—1X (14x18 pixels) or 2X (28x36 pixels). Inter-row spacing is programmable from 0 to 15 horizontal scan lines. This allows user to create pseudo icons using multiple characters with 0-row spacing.

A 14-bit Pulse Width Modulator (PWM) port provides enough voltage resolution for a voltage synthesizer tuning system. Ten 6-bit PWM ports are used for controlling audio (base, treble, balance, and volume) and video (contrast, brightness, color, tint, and sharpness) signal levels.

There are 27 I/O pins dedicated to input and output functions. They are grouped into four ports, and are configurable under software control to provide timing, status signals, serial and parallel I/O.

To handle real-time events, such as counting, timing, and data communication, two on-chip counter/timers with a large number of user-selectable modes are implemented.

The device is housed in a 42-pin SDIP and provides an ideal, reliable solution for high-volume consumer television applications.

Table 1-1. Z90200-Family Product Summary

Device	Memory	I^2C	CCD & V-Chip	Pins, Package	ADC	PWM Ports		Timers	IR
						14 Bit	6 Bit		
Z90209	ICE	No	No	124, PGA	3 bit,4 Channels	1	10	2	Yes
Z90202	12 KB ROM	No	No	42, SDIP	3 bit, 4 Channels	1	10	2	Yes
Z90203	16 KB ROM	No	No	42, SDIP	3 bit, 4 Channels	1	10	2	Yes
Z90219	ICE	Yes	No	124, PGA	3 bit, 4 Channels	1	10	2	Yes
Z90211	OTP	Yes	No	42, SDIP	3 bit, 4 Channels	1	10	2	Yes
Z90229	ICE	Yes	Yes	124, PGA	4 bit, 4 Channels	1	10	2	Yes
Z90221	OTP	Yes	Yes	42, SDIP	4 bit, 4 Channels	1	10	2	Yes
Z90224	24 KB ROM	Yes	Yes	42, SDIP	4 bit, 4 Channels	1	10	2	Yes
Z90239	ICE	Yes	No	124, PGA	4 bit, 4 Channels	1	10	2	Yes
Z90231	OTP	Yes	No	42, SDIP	4 bit, 4 Channels	1	10	2	Yes
Z90232	12 KB ROM	Yes	No	42, SDIP	4 bit, 4 Channels	1	10	2	Yes
Z90233	16 KB ROM	Yes	No	42, SDIP	4 bit, 4 Channels	1	10	2	Yes
Z90234	24 KB ROM	Yes	No	42, SDIP	4 bit, 4 Channels	1	10	2	Yes

CHAPTER 2
ARCHITECTURAL OVERVIEW

2.1 INTRODUCTION

The Z90239 Digital Television Controller functions as the result of the interaction between hardware and software. A series of registers stores settings for the On-Screen Display that is output through the hardware device.

2.2 HARDWARE

Two formats are used for this family of devices. The Z90239 in a 124-Pin PGA ceramic package is used with the ICEBOX Emulator during design and debugging. OTP and production devices utilizes a 42-pin SDIP format. Pin identification and assignments are provided below for both formats.

2.2.1 Pin Identification

Figure 2-1 shows the pin numbers for the OTP and production device format. Following the figure, Table 2-1 describes the function that each pin is assigned to.

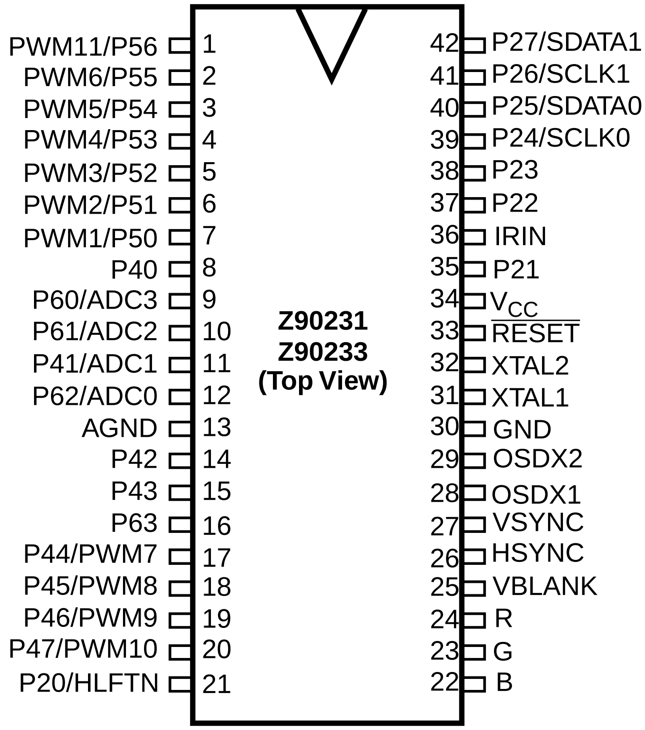


Figure 2-1. Z90231 and Z90233 Pin Identification

Note: The pins on the Z90230 are assigned to perform the functions identified in Table 2-1.

Note: In this and the following sections, all Signals with an overbar are active Low.

Table 2-1. Z90230-Family OTP and Production Pin Assignment

Name	Pin Function	Package 42-Pin SDIP	Direction	POR
V_{CC}	+5 Volts	34	Power	Power
GND, AGND	0 Volts	30, 13	Power	Power
IRIN	Infrared Remote Capture Input	36	I	I
PWM11	14-bit Pulse Width Modulator Output	1	O	N/A
PWM10-PWM1	6-Bit Pulse Width Modulator Output	20, 19, 18, 17, 2, 3, 4, 5, 6, 7	O	N/A
P5 (6-0)	Bit Programmable I/O Ports	1, 2, 3, 4, 5, 6, 7	I/O	I
P2 (7-0)	Bit-Programmable I/O Ports	42, 41, 40, 39, 38, 37, 35, 21	I/O	I
HLFTN	Halftone Output	21	O	N/A
SDATA0, SDATA1	I^2C Data, Bidirectional (Send/Receive) Serial Data Lines	40, 42[1]	I/O	N/A
SCLK0, SCLK1	I^2C Clock	39, 41[1]	I/O	N/A
P6 (3-0)	Bit-Programmable I/O Ports	16, 12, 10, 9	I/O	I
P4 (7-0)	Bit-Programmable I/O Ports	20, 19, 18, 17, 15, 14, 11, 8	I/O	I
XTAL1	Crystal Oscillator Input	31	I	I
XTAL2	Crystal Oscillator Output	32	O	O
OSDX1	Dot-Clock Oscillator Input	28	I	I
OSDX2	Dot-Clock Oscillator Output	29	O	O
H_{SYNC}	Horizontal Synchronization	26	I	I
V_{SYNC}	Vertical Synchronization	27	I	I
VBLANK	Video Blanking	25	O	O
R,G,B	Video Red, Green, Blue	24, 23, 22	O	O
ADC3-ADC0	4-Bit Analog-to-Digital Converter Input	9, 10, 11, 12	AI	I
$\overline{\text{RESET}}$	Device Reset	33	I/O	I

Note:
1. When Pins 39-42 are configured for I^2C, pins 39 and 40 comprise one channel, and pins 41 and 42 comprise another channel.

2.2.2 Z90239 124-Pin PGA Ceramic Package Pin-Out Diagram

The Z90239 ICE chip contains more pins than the production devices. The additional pins provide internal values that are valuable during design activities.

Figure 2-2 illustrates the pin assignment of the Z90239 ICE chip. Following the figure, Table 2-2 describes the assignment for each pin.

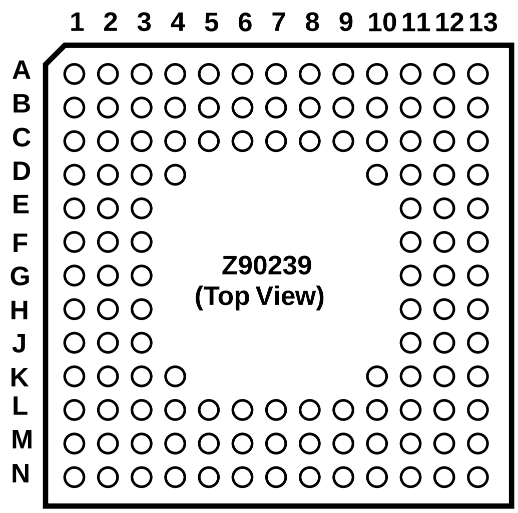


Figure 2-2. Z90239 124-Pin PGA Ceramic Package Pin-Out Diagram

2.2.3 Z90239 Pin Assignment

The function of each pin of the Z90239 ICE chip is described in the table below.

Table 2-2. Z90239 Pin Assignments

Name	Pin Function	42-Pin SDIP	Package 124-Pin QFP	Direction	Power
NC			C3, B2, B1		
OSDX1	OSD Dot Clock Oscillator Input	28	D3	I	I
OSDX2	OSD Dot Clock Oscillator Output	29	C2	O	O
GND	Ground	30	C1	Power	Power
MADR14-MADR0	MCU Address		D2, E3, D1, F3, F2, F1, G2, G3, G1, H1, H2, H3, J1, J2, J3	O	O
XTAL1	External Crystal Oscillator	31	E2	I	I

Table 2-2. Z90239 Pin Assignments (Continued)

Name	Pin Function	42-Pin SDIP	Package 124-Pin QFP	Direction	Power
XTAL2	External Crystal Oscillator	32	E1	O	O
$\overline{\text{RESET}}$	System Reset	33	K1	I/O	I
V_{CC}	Power Supply	34	K2	Power	Power
$\overline{\text{SYNC}}$	Last T Cycle		L1	O	O
P21	Port 2 Pin 1	35	M1	I/O	I
NC			K3, L2, N1, K4, L3, M2		
IRIN	IR Serial Data Input	36	N2	I	I
P22, P23	Port 2 Pin 2, 3	37, 38	L4, M3, N3	I/O	I
P24/SCLK0	Port 2 Pin 4/I^2C Clock	39	N3	I/O	I
$\overline{\text{MAS}}$	MCU Address Strobe		M4	O	O
P25/SDATA0	Port 2 Pin 5/I^2C Data	40	L5	I/O	I
P26/SCLK1, P27/SDATA1	Port 2 Pin 6, 7/I^2C Clock, Data	41, 42	M5, M6	I/O	I
$\overline{\text{MDS}}$	MCU Data Strobe		N4	O	O
CGDATA6-CGDATA0	CGROM Data		N5, L6, N6, M7, L7, N7, N8	I	I
PWM11	Pulse Width Modulator 11/Port 5 Pin 6	1	M8	O	O
V_{DD}	Power Supply		L8, N9	Power	Power
PWM6/P55-PWM1/P50	Pulse Width Modulator/Port 5 Pin 5, 4, 3, 2, 1, 0	2, 3, 4, 5, 6, 7	M9, L9, N11, N12, L10,M11	I/O	I
SCLK	System Clock		N10	O	O
NC			N13, K10, L11, M12, M13, K11		
P40	Port 4 Pin 0	8	L12	I/O	I
P60/ADC3	Port 6 Pin 0/ADC3	9	L13	I/O	I

Table 2-2. Z90239 Pin Assignments (Continued)

Name	Pin Function	42-Pin SDIP	Package 124-Pin QFP	Direction	Power
CGADR0-CGADR13	CGROM Address		K12, K13, J12, J13, H11, H12, H13, G12, G13, F13, F12, F11, E12, D13	O	O
P61/ADC2	Port 6 Pin 1/ADC2	10	J11	I/O	I
P41/ADC1	Port 4 Pin 1/ADC1	11	G11	I/O	I
P62/ADC0	Port 6 Pin 2/ADC0	12	E13	I/O	I
SIZE	0-16 KB System ROM 1-32 KB System ROM		E11	I	I
AGND	Analog Ground	13	D12	Power	Power
P42, P43	Port 4 Pin 2,3	14, 15	C13, A12	I/O	I
NC			B13, D11, C12, A13, D10, C11, B12, M10		
P63	Port 6 Pin 3	16	C10	I/O	I
P44-P47/PWM7-PWM10	Port 4 Pin 4, 5,6, 7/ PWM 7, 8, 9,10	17, 18, 19, 20	B11, A11, C9, B9	I/O	I
ICE	External ROM Selection		B10	I	I
$\overline{\text{DTIMER}}$	Disable WDT/Timer0, Timer1		A10	I	I
GND	Ground		A9	Power	Power
P20/HLFTN	Port 2 Pin 0/Halftone Output	21	C8	I/O	I
MDATA0-MDATA7	MCU Data		B8, A8, B7, C7, A7, B6, C6, B5	I	I
B	Blue (Video)	22	A6	O	O
G	Green (Video)	23	A5	O	O
R	Red (Video)	24	A4	O	O
NC			C5		
VBLANK	Video Blank	25	B4	O	O

Table 2-2. Z90239 Pin Assignments (Continued)

Name	Pin Function	42-Pin SDIP	Package 124-Pin QFP	Direction	Power
H_{SYNC}	Horizontal Synchronization	26	A3	I	I
V_{SYNC}	Vertical Synchronization	27	A2	I	I
$\overline{IACK}$	Interrupt Acknowledge		C4	O	O
NC			B3, A1, D4		

2.2.4 Pin Descriptions

2.2.4.1 Single-Purpose Pin Descriptions

AGND *Analog Ground.*

B *Blue.* CMOS output of the blue video signal B-Y. Video blue is programmable for either polarity.

CGADR0-CGADR13 *CGROM Addresses 0 through 13.*

CGDATA6-CGDATA0 *CGROM Data Input Pins 6 through 0.*

G *Green.* CMOS output of the green video signal G-Y. Video green is programmable for either polarity.

GND *Ground.*

H_{SYNC} *Horizontal Sync.* Pin input for external horizontal synchronization signal.

IACK *Interrupt Acknowledge.*

ICE *External ROM Selection.*

IRIN *Infrared Capture Input.*

MAS *MCU Address Strobe Output.*

MDS *MCU Data Strobe.*

MDATA0-MDATA7 *MCU Data Input Bits 0 through 7.*

MADR14-MADR0 *MCU Address Output Bits 14 through 0.*

NC *No Connection.*

OSDX1, OSDX2 *On-Screen Display Dot Clock Oscillators OSDX1 and OSDX2.* These oscillator input and output pins for on-screen display circuits are connected to an inductor and two capacitors to generate the character dot clock. The dot clock frequency determines the character pixel width and phase synchronized to H_{SYNC}.

P21 *Port 2 Pin 1.*

P22, P23 *Port 2 Pins 2 and 3.*

P40 *Port 4 Pin 0.* Bidirectional digital port, configured to read digital data or to send output to an attached device. This pin is not multiplexed.

P42, 43 *Port 4 Pins 2 and 3.*

R *Red.* CMOS output of the red video signal R-Y. Video red is programmable for either polarity.

RESET *System Reset.*

SCLK *System Clock.*

SIZE *System ROM Size.* When the value is 0, available system ROM is 16 KB. When the value is 1, available system ROM is 32 KB.

SYNC *Last Timer Cycle.*

VBLANK *Video Blank.* CMOS output, programmable polarity. This pin is used as a superimpose control port to display characters from video RAM. The signal controls Y-signal output of CRTs and turns off the incoming video display while the characters in video RAM are superimposed on the screen. The output ports of color data directly drive three electron guns on the CRT at the same time VBLANK output turns off the Y signal.

V_{CC} *Power Supply.*

V_{DD} *Power Supply.*

V_{SYNC} *Vertical Sync*. Pin input for external vertical synchronization signal.

XTAL1, XTAL2 *Time-Based Input, Output respectively.* These pins connect to the internal parallel-resonant clock crystal oscillator circuit with two capacitors to GND. XTAL1 can be used as an external clock input. Low EMI noise operation deletes a divide-by-2 in the instruction clock timing chain.

2.2.4.2 Multiplexed Pin Descriptions

DTIMER *Disable Watch-Dog Timer or Timers 0 and 1.*

P20/HLFTN *Port 2 Pin 0 or Halftone Output.* Port 2 is 8-bit, CMOS compatible, and each bit is programmable for either input or output. Input buffers are Schmitt triggered. Bits programmable as outputs may be globally programmed as either push-pull or open drain. Port operation is accomplished by Port 2 Mode Register at F6h. Port 2 is at 02h, which is part of the Register File.

P24/SCLK0 *Port 2 Pin 4 or I^2C Clock.*

P25/SDATA0 *Port 2 Pin 5 or I^2C Data.*

P26/SCLK1, P27/SDATA1 *Port 2 Pin 6 or I^2C Clock, and Port 2 Pin 7 or I^2C Data.*

P62/ADC0 *Port 6 Pin 2 or Analog-to-Digital Converter Channel 0.* P62 may be read directly. A negative edge event is latched into IRQ 2 to initiate an IRQ 2-vectored interrupt, if appropriately enabled.

P60/ADC3 *Port 6 Pin 0 or Analog-to-Digital Converter Channel 3.* Port 6 pin 0 is a programmable input or output line.

P61/ADC2 *Port 6 Pin 1 or Analog-to-Digital Converter Channel 2.* Port 6 pin 1 is a programmable input or output line.

P41/ADC1 *Port 4 Pin 1or Analog-to-Digital Converter Channel 1.*

P63 *Port 6 Pin 3.* P63 input may be read directly at 03h. A negative edge event is latched to IRQ 3. An IRQ3-vectored interrupt occurs if appropriately enabled. A typical application would place the device in Stop mode when P63 goes Low (IRQ 3 interrupt routine). When P63 subsequently goes High, a Stop-Mode Recovery is initiated.

P44/PWM7 *Port 4 Pin 4 or Pulse Width Modulator 7.* Port 4 pin 4 is a programmable input or output port. The PWM channel has 6-bit resolution.

P45/PWM8 *Port 4 Pin 5 or Pulse Width Modulator 8.* Port 4 pin 5 is a programmable input or output port. The PWM channel has 6-bit resolution.

P46/PWM8 *Port 4 Pin 6 or Pulse Width Modulator 9.* Port 4 pin 6 is a programmable input or output port. The PWM channel has 6-bit resolution.

P47/PWM10 *Port 4 Pin 7 or Pulse Width Modulator 10.* Port 4 pin 7 is a programmable input or output port. The PWM channel has 6-bit resolution.

PWM11/P56 *Pulse Width Modulator 11 or Port 5 Pin 6.* The PWM signal-generator channel has 14-bit resolution. Port 5 pin 6 is a programmable input or output port.

PWM6/P55 *Pulse Width Modulator 6 or Port 5 Pin 5.* The PWM signal-generator channel has 6-bit resolution. Port 5 pin 5 is a programmable input or output port.

PWM5/P54 *Pulse Width Modulator 5 or Port 5 Pin 4.* The PWM signal-generator channel has 6-bit resolution. Port 5 pin 4 is a programmable input or output port.

PWM4/P53 *Pulse Width Modulator 4 or Port 5 Pin 3.* The PWM signal-generator channel has 6-bit resolution. Port 5 pin 3 is a programmable input or output port.

2.2.5 Core Customization

Several features have been added to and removed from the internal microprocessor used in the Z86C43 to form the Z90230 family. However, the description of core still applies to the Z90210 and Z90230 DTC family of applications. Information about the registers is included in Chapter 3.

The following Z86C43 features are not available in the Z90230 family:

- P3 voltage comparators are not supported.
- Port handshaking is not supported.
- Port 0 and Port 1 are not available, and yield 0s when read.

PWM3/P52 *Pulse Width Modulator 3 or Port 5 Pin 2.* The PWM signal-generator channel has 6-bit resolution. Port 5 pin 2 is a programmable input or output port.

PWM2/P51 *Pulse Width Modulator 2 or Port 5 Pin 1.* The PWM signal-generator channel has 6-bit resolution. Port 5 pin 1 is a programmable input or output port.

PWM1/P50 *Pulse Width Modulator 1 or Port 5 Pin 0.* The PWM signal-generator channel has 6-bit resolution. Port 5 pin 0 is a programmable input or output port.

- P32 and P33 port interrupts are not available, and yield 0s when read.
- WDT is not clocked when in Stop Mode.
- Timer 1 is used for horizontal synchronization; P62 input is no longer valid as the external clock to Timer1. (P62 is still an interrupt input port.)
- P62 edge selection in interrupt request register has been modified.

2.2.6 Block Diagram

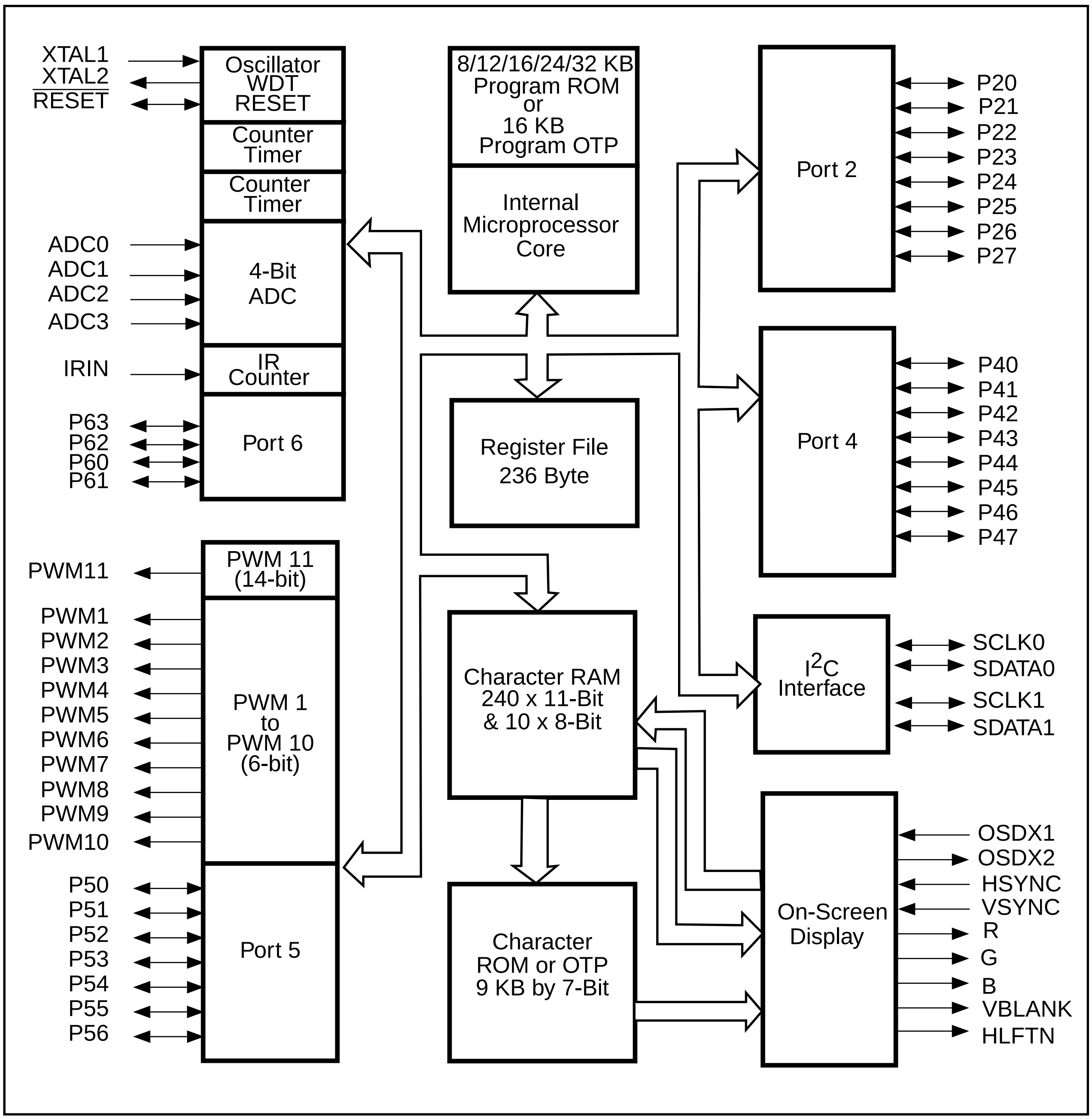


Figure 2-3. Block Diagram

2.3 CONTROL REGISTERS

Most of the control registers are mapped into expanded register file groups in the internal microprocessor core. Refer to the *Z8 Microcontrollers User's Manual* for a detailed functional description.

2.3.1 3-Bit ADC Data Register

Four multiplexed analog inputs are available to either a 3-bit or 4-bit analog-to-digital converter (ADC) depending on the configuration.

Figure 2-4, Figure 2-5, and Figure 2-6 describe the 3ADC_DTA, 4ADC_DTA, and PIN_SLT registers for ADC control and I/O mode selections:

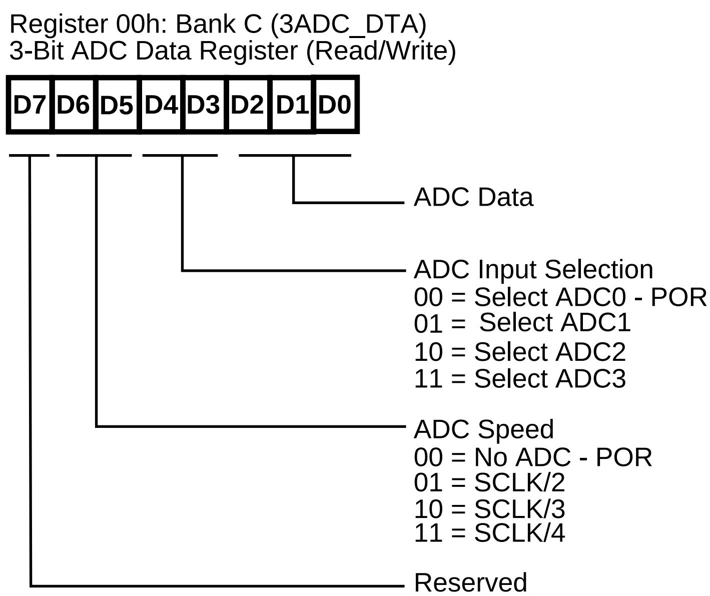


Figure 2-4. 3-Bit ADC Data Register

2.3.2 4-Bit ADC Data Register

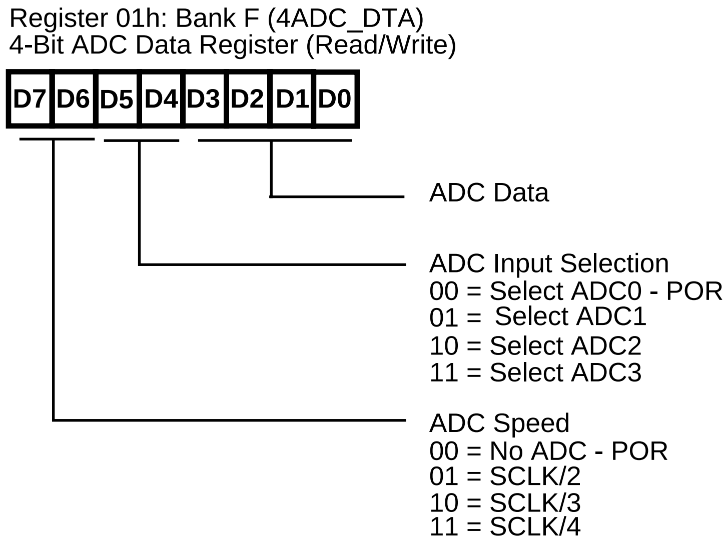


Figure 2-5. 4-Bit ADC Data Register

P41 must be set to input mode for ADC 1 selection.

2.3.3 Port 4 Pin-Out Selection Register

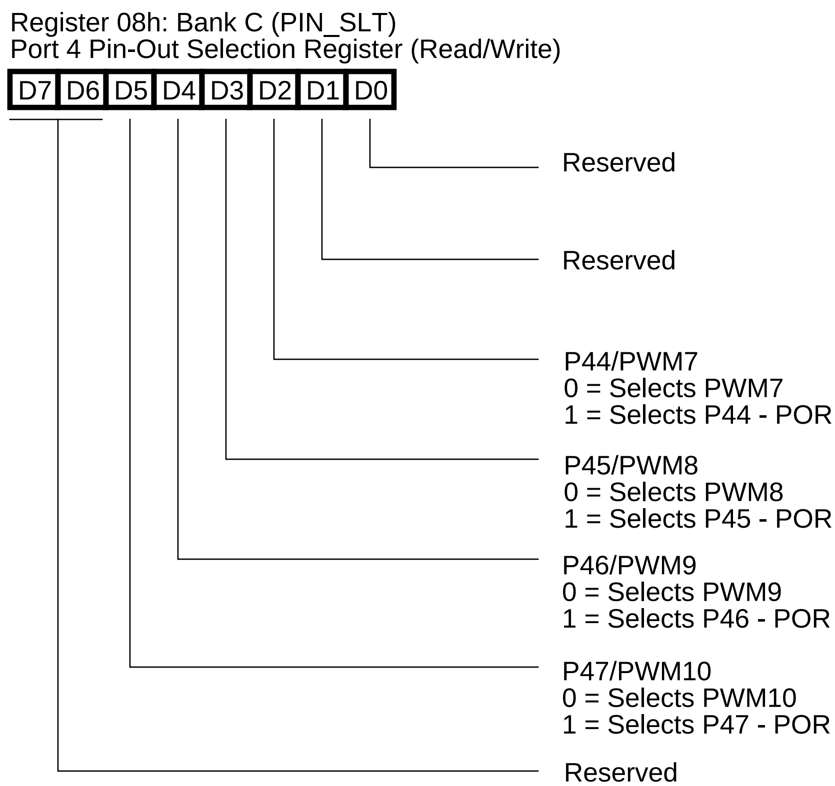
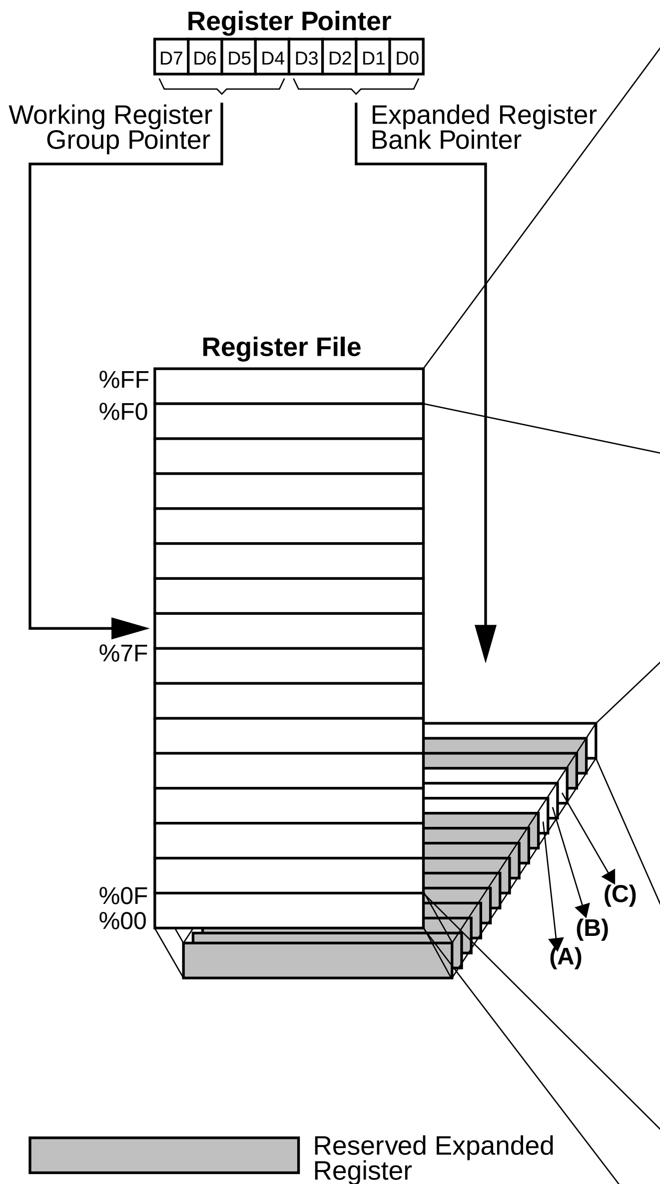


Figure 2-6. Port 4 Pin-Out Selection Register

2.3.4 Expanded Register File



Register		D7	D6	D5	D4	D3	D2	D1	D0
%FF	SPL	U	U	U	U	U	U	U	U
%FE	SPH	U	U	U	U	U	U	U	U
%FD	RP	0	0	0	0	0	0	0	0
%FC	FLAGS	U	U	U	U	U	U	U	U
%FB	IMR	0	U	U	U	U	U	U	U
%FA	IRQ	0	0	0	0	0	0	0	0
%F9	IPR	U	U	U	U	U	U	U	U
%F8	P01M	1	1	1	1	1	1	1	1
%F7	P2CNTL	0	0	0	0	0	0	0	1
%F6	P2M	1	1	1	1	1	1	1	1
%F5	PRE0	U	U	U	U	U	U	U	0
%F4	T0	U	U	U	U	U	U	U	U
%F3	PRE1	U	U	U	U	U	U	0	0
%F2	T1	U	U	U	U	U	U	U	U
%F1	TMR	0	0	0	0	0	0	0	0
%F0	Reserved								

Expanded Register Bank (F)

Register		Reset Condition							
%(F)0F	WDTMR	U	U	U	0	1	1	0	1
%(F)0E	Reserved								
%(F)0D	Reserved								
%(F)0C	Reserved								
%(F)0B	SMR	0	0	1	0	0	0	0	0
%(F)0A	Reserved								
%(F)09	Reserved								
%(F)08	Reserved								
%(F)07	MC_Reg	0	0	0	0	0	0	0	0
%(F)06	MR_En	0	0	0	0	0	0	0	0
%(F)05	MC_End	U	U	U	0	0	0	0	0
%(F)04	MC_St	U	U	U	0	0	0	0	0
%(F)03	PRT6_DTA	U	U	U	U	1	1	1	1
%(F)02	PRT6_DRT	1	1	1	1	1	1	1	1
%(F)01	4ADC_DTA	0	0	0	0	U	U	U	U
%(F)00	PCON	1	1	1	1	1	1	1	0

Register		Reset Condition							
%(0)03	Reserved								
%(0)02	P2	U	U	U	U	U	U	U	U
%(0)01	Reserved								
%(0)00	Reserved								

Figure 2-7. Register and Expanded Register File Map

Bank A

Register	Address
OSD Control (OSD_CNTL)	00h
Vertical Position (VERT_POS)	01h
Horizontal Position (HOR_POS)	02h
Display Attribute (DISP_ATTR)	03h
Row Space (ROW_SPACE)	04h
Fade Position 1 (FADE _POS)	05h
Fade Position 2 (FADE_POS)	06h
Second Color Control (SNDCLR_CNTRL)	07h
Second Color (SNDCLR)	08h
Color Palette 0 (CLR_P0)	09h
Color Palette 1 (CLR_P1)	0Ah
Color Palette 2 (CLR_P2)	0Bh
Color Palette 3 (CLR_P3)	0Ch
Color Palette 4(CLR_P4)	0Dh
Color Palette 5 (CLR_P5)	0Eh
Color Palette 6 (CLR_P6)	0Fh

Bank B

Register	Address
PWM11-High Data Register (PWM11H)	00h
PWM11-Low Data Register (PWM11L)	01h
PWM 1 Data Register (PWM1)	02h
PWM 2 Data Register (PWM2)	03h
PWM 3 Data Register (PWM3)	04h
PWM 4 Data Register (PWM4)	05h
PWM 5 Data Register (PWM5)	06h
PWM 6 Data Register (PWM6)	07h
PWM 7 Data Register (PWM7)	08h
PWM 8 Data Register (PWM8)	09h
PWM 9 Data Register (PWM9)	0Ah
PWM 10 Data Register (PWM10)	0Bh
Port 5 Data Register (PRT5_DTA)	0Ch
PWM MODE Register (P_MODE)	0Dh
Port 5 Direction Register (PRT5_DRT)	0Eh

Bank C

Register	Address
3-Bit ADC Data Register (3ADC_DTA)	00h
Timer Control Register 0 (TCR0)	01h
Timer Control Register 1 (TCR1)	02h
IR Capture Register 0 (IR_CP0)	03h
IR Capture Register 1(IR_CP1)	04h
Port 4 Data Register (PRT4_DTA)	05h
Port4 Direction Register(PRT4_DRT)	06h
HVInterrupt Status Register (INT_ST)	07h
Port 4 Pin-Out Selection (PIN_SLT)	08h
Color Index Register (CLR_IDX)	09h
Master I2C Data Register (I2C_DATA)	0Ah
I2C Command Register (I2C_CMD)	0Bh
Master I2C Control Register(I2C_CNTL)	0Ch

Bank F

Register	Address
Port Configuration Register (PCON)	00h
4-Bit ADC Data Register (4ADC_DTA)	01h
Port 6 Direction Control Register (PRT6_DRT)	02h
Port 6 Data Register (PRT6_DTA)	03h
Mesh Column Start (MC_St)	04h
Mesh Column End (MC_End)	05h
Mesh Row Enable (MR_En)	06h
Mesh Control Register (MC_Reg)	07h
N/A	08h
N/A	09h
N/A	0Ah
Stop-Mode Recovery Register (SMR)	0Bh
N/A	0Ch
N/A	0Dh
N/A	0Eh
WDT Mode Register (WDTMR)	0Fh

Figure 2-8. Expanded Register File

2.3.5 Stop-Mode Recovery Register

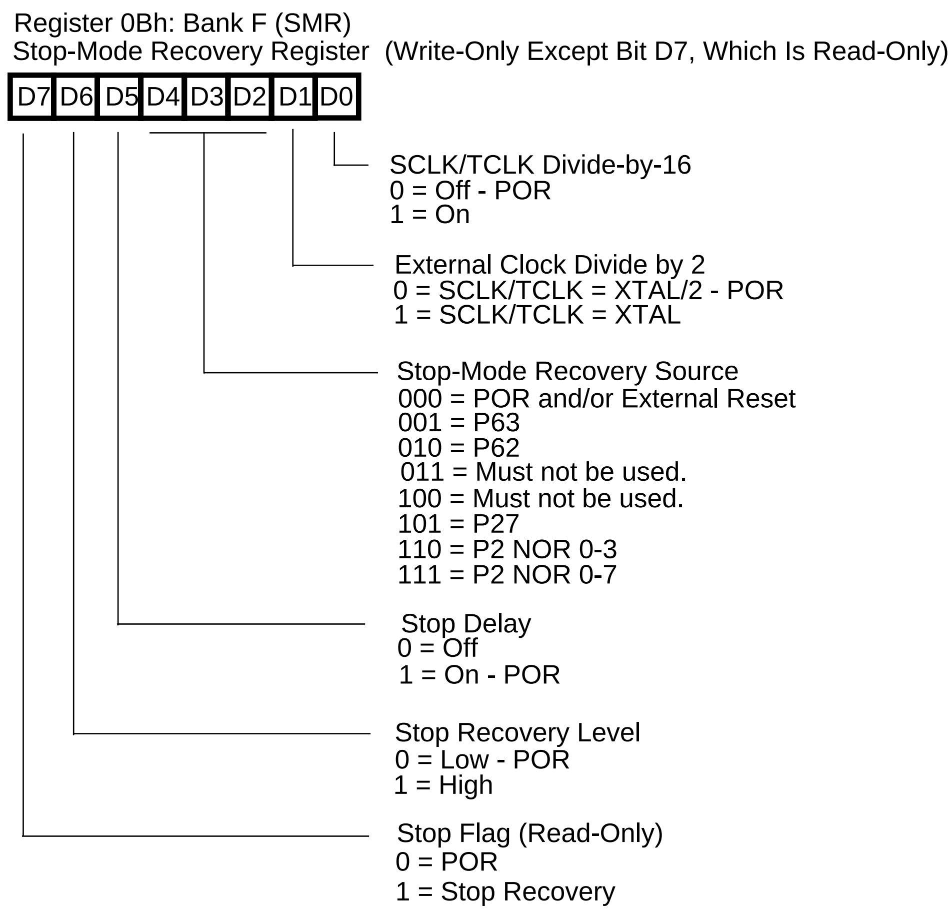


Figure 2-9. Stop-Mode Recovery Register

Note: The Stop-Mode Recovery Source values 011 and 100 are reserved and must not be used.

2.3.6 Watch-Dog Timer Mode Register

The WDT always uses the internal RC oscillator.

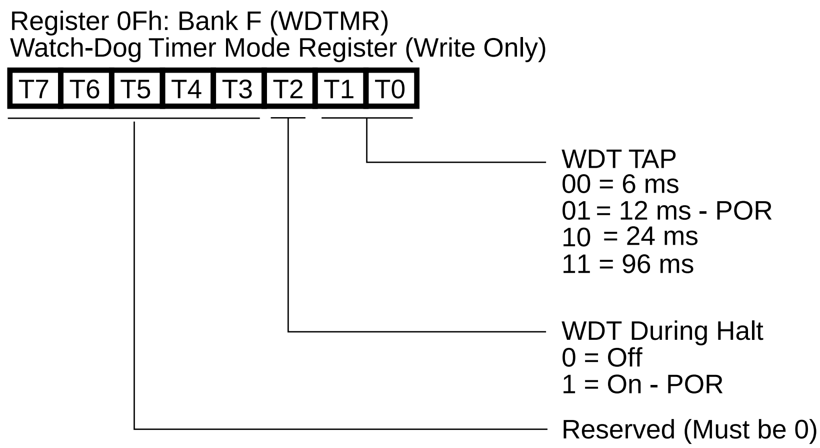


Figure 2-10. Watch-Dog Timer Mode Register

2.3.7 Timer Mode Register

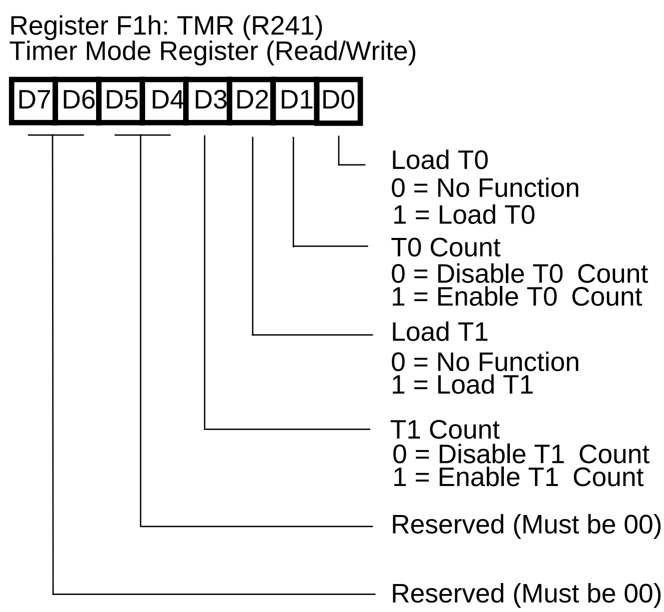


Figure 2-11. Timer Mode Register

2.3.8 Counter/Timer 1 Register

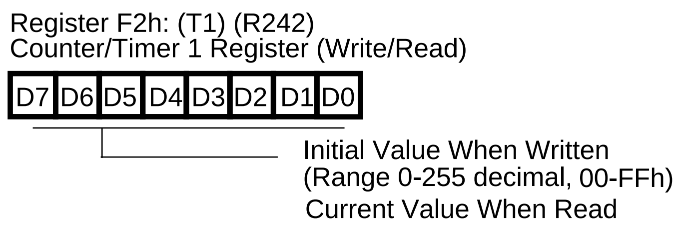


Figure 2-12. CounterTimer1 Register

2.3.9 Prescaler 1 Register

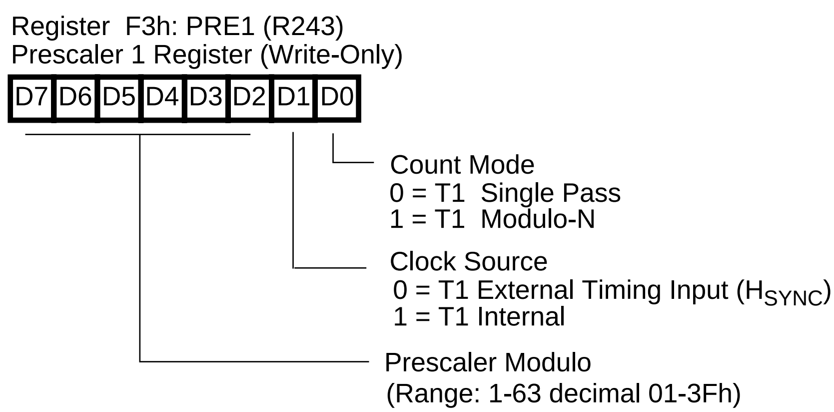


Figure 2-13. Prescaler 1 Register

2.3.10 Counter/Timer 0 Register

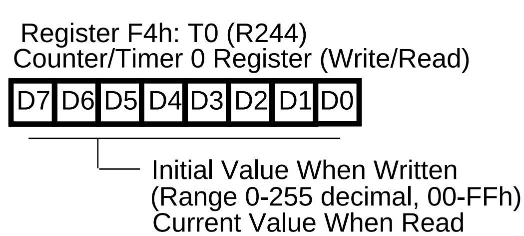


Figure 2-14. Counter/Timer 0 Register

2.3.11 Prescaler 0 Register

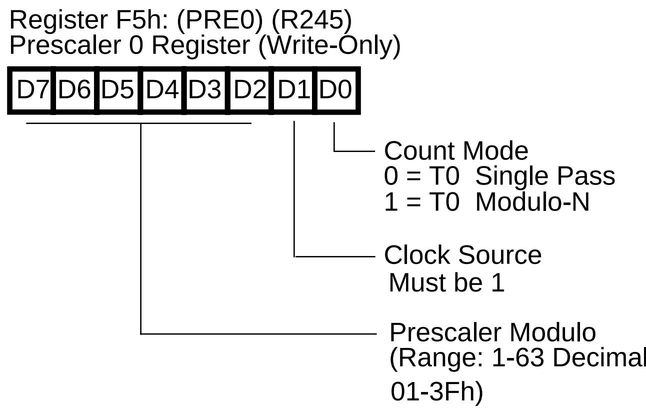


Figure 2-15. Prescaler 0 Register

2.3.12 Port 2 Mode Register

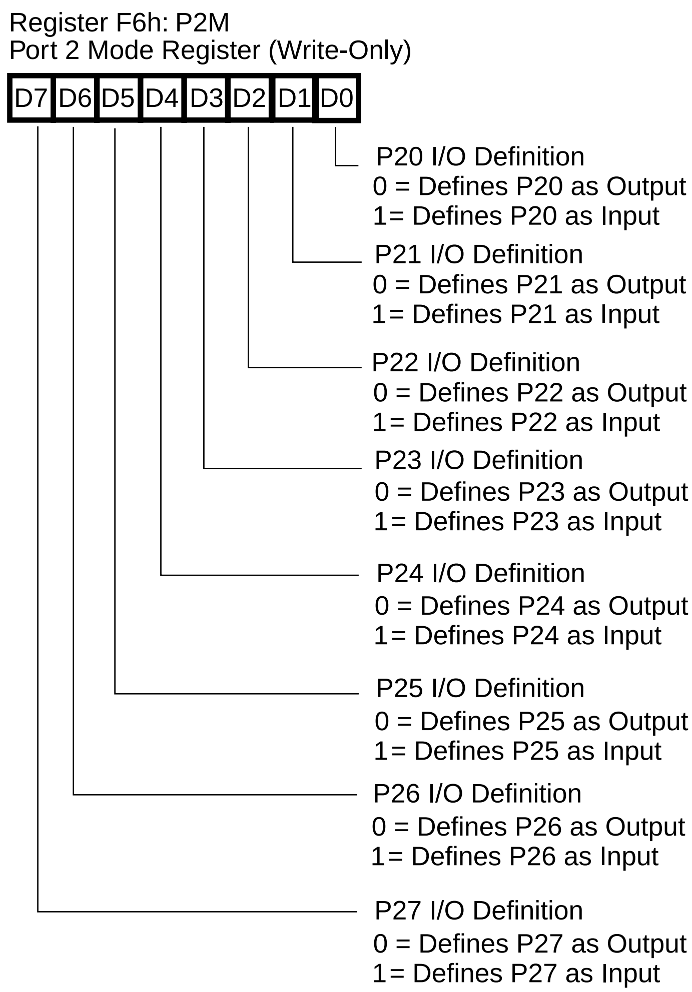


Figure 2-16. Port 2 Mode Register

2.3.13 Port 2 Control Register

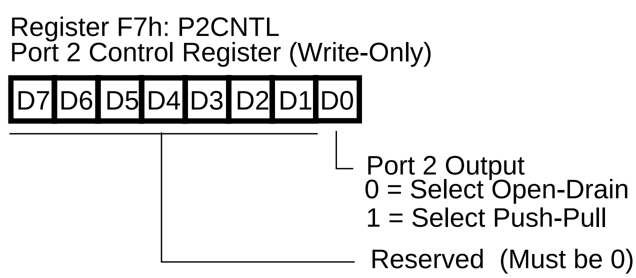


Figure 2-17. Port 2 Control Register

If P27/P26 are selected as I^2C channel 1 or P25/P24 are selected as I^2C channel 0, then selected pins in the I^2C channel are automatically set into open-drain mode regardless of the setting in this control register. If P20 is used as a halftone pin, then this pin becomes push-pull regardless of the setting in this control register.

2.3.14 Interrupt Priority Register

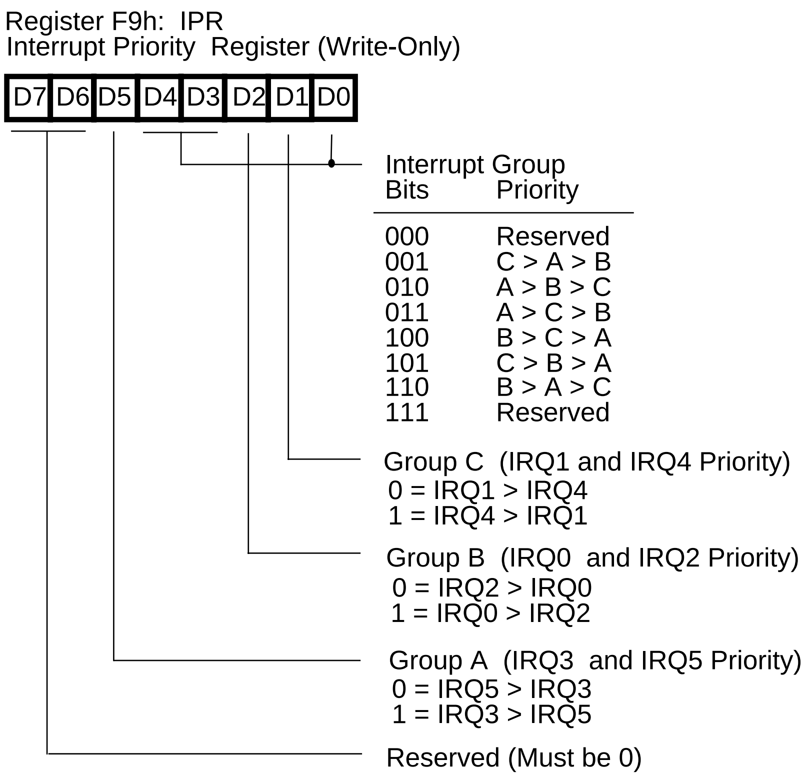


Figure 2-18. Interrupt Priority Register

Whenever Power-On Reset (POR) is executed, the IRQ register is reset to 00h and the interrupt state machine is disabled. Before the IRQ Register can accept requests, the IRQ register must be enabled by executing an Enable Interrupts (EI) instruction.

2.3.15 Interrupt Request Register

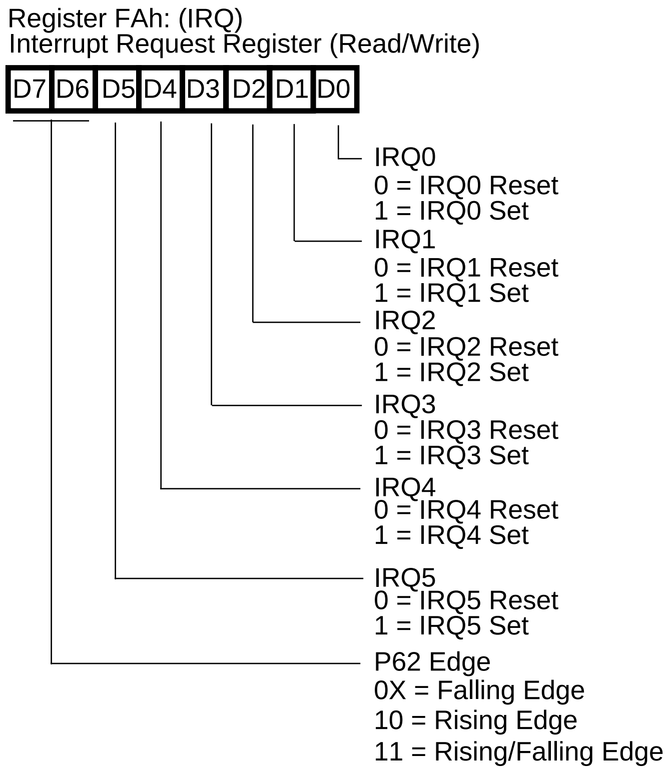


Figure 2-19. Interrupt Request Register

The functions of the IRQs are as follows:

Table 2-3. IRQ Function Summary

IRQ	Function
IRQ0	IR Input
IRQ1	HV_{SYNC} Input
IRQ2	P62 Input
IRQ3	P63 Input
IRQ4	T0 Internal Timer
IRQ5	T1 Internal Timer

Note: P62 and P63 must be configured as input if used as an interrupt source.

Data bits 6 and 7 set the P62 edge. Some coding is required to clear P62 for input:

- To select “Rising Edge” for P62 interrupt:

```
.....
.....
DI                   ; disable all interrupts
OR IRQ    #%80       ; enable rising edge for P62
                     ; interrupt
AND IRQ   #%FB       ; clear IRQ2 (P62 interrupt),
                     ; keep other IRQs' bits
                     ; untouched
EI                   ; enable interrupts
.....
.....
```

- To select “Rising & Falling Edge” for P62 interrupt:

```
.....
.....
DI                   ; disable all interrupts
OR IRQ    #%C0       ; enable rising & falling edge
                     ; for P62 interrupt
AND IRQ   #%FB       ; clear IRQ2 bit (P62 |
                     ; interrupt), keep other IRQ's
                     ; bits untouched
EI                   ; enable interrupts
.....
.....
```

The IMR is cleared before the IRQ enabling sequence to insure no unexpected interrupts occur when EI is executed. This code sequence should be executed prior to programming the application required values for IPR and IMR.

Note: IRQ bits 6 and 7 are device dependent. When reserved, the bits are not used and will return a 0 when read. When used as the Interrupt Edge select bits, the configuration options are as shown in the following table.

Table 2-4. IRQ Register Configuration

IRQ		Interrupt Edge
D7	**D6**	**P62**
0	0	Falling
0	1	Falling
1	0	Rising
1	1	Rising/Falling

The proper sequence for programming the interrupt edge select bits is shown in the following

Assembly code (assumes IPR and IMR have been previously initialized):

```
DI                          ;Inhibit all
                            ;interrupts
                            ;until input edges are
                            ;configured
OR   IRQ,#XX 000000B        ;Configure interrupt
                            ;do not disturb
                            ;edges as needed -
                            ;IRQ 0-5.
EI                          ;Re-enable interrupts.
```

2.3.16 Interrupt Mask Register

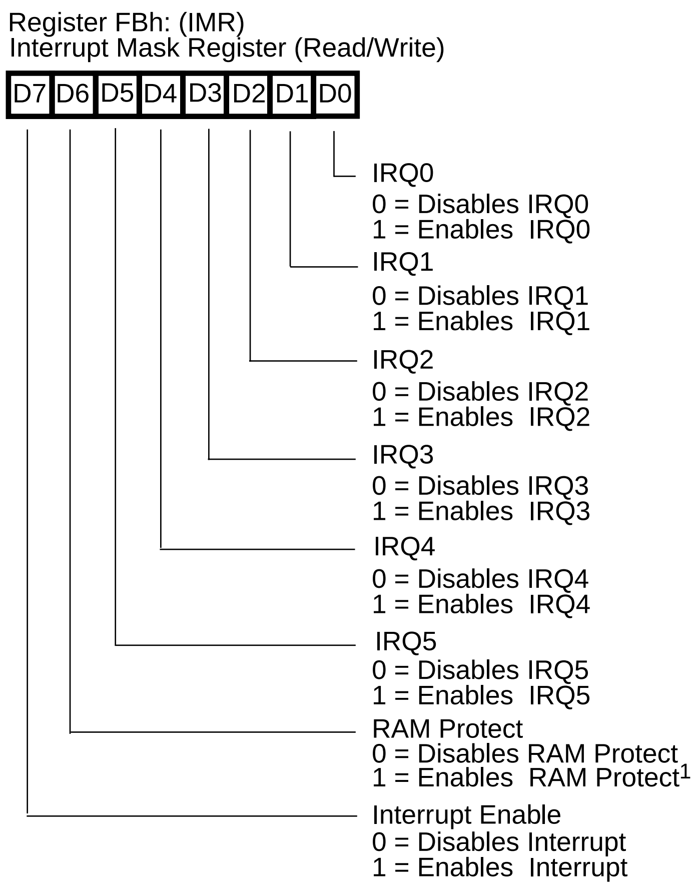


Figure 2-20. Interrupt Mask Register

Note:
1. This option must be selected when ROM code is submitted for ROM masking. Otherwise, this control bit is disabled permanently.

2.3.17 Flags Register

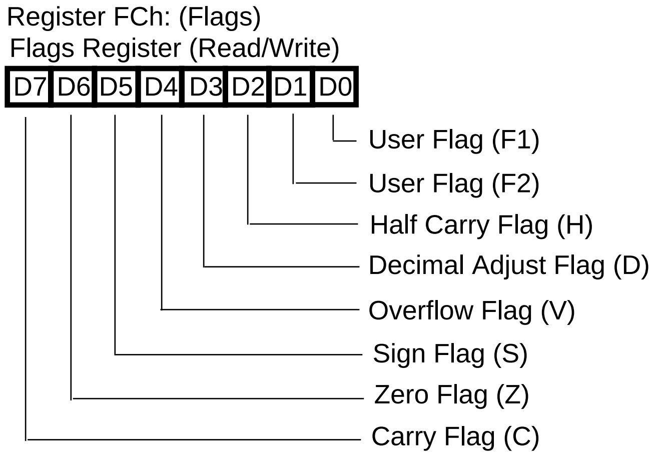


Figure 2-21. Flags Register

2.3.18 Register Pointer

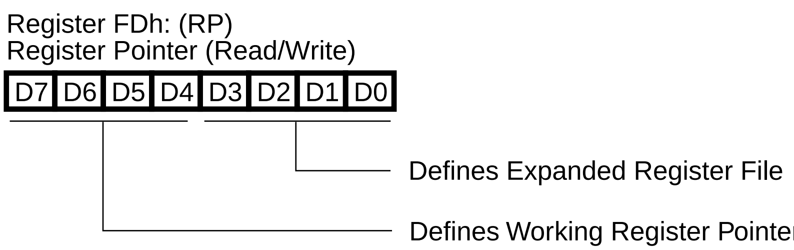


Figure 2-22. Register Pointer

2.3.19 Stack Pointer High

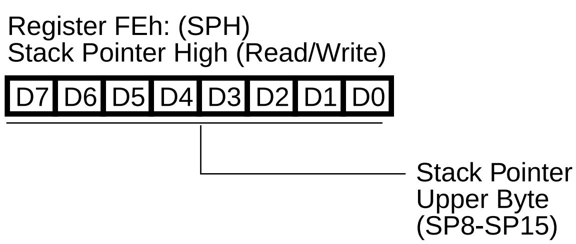


Figure 2-23. Stack Pointer High Register

There are 236 (FFh) general-purpose registers in the Z90230 family of products. The SPH register is reserved for future expansion.

2.3.20 Stack Pointer Low

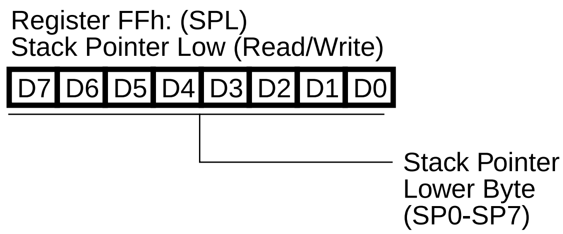


Figure 2-24. Stack Pointer Low Register

2.3.21 Port 2 Data Register

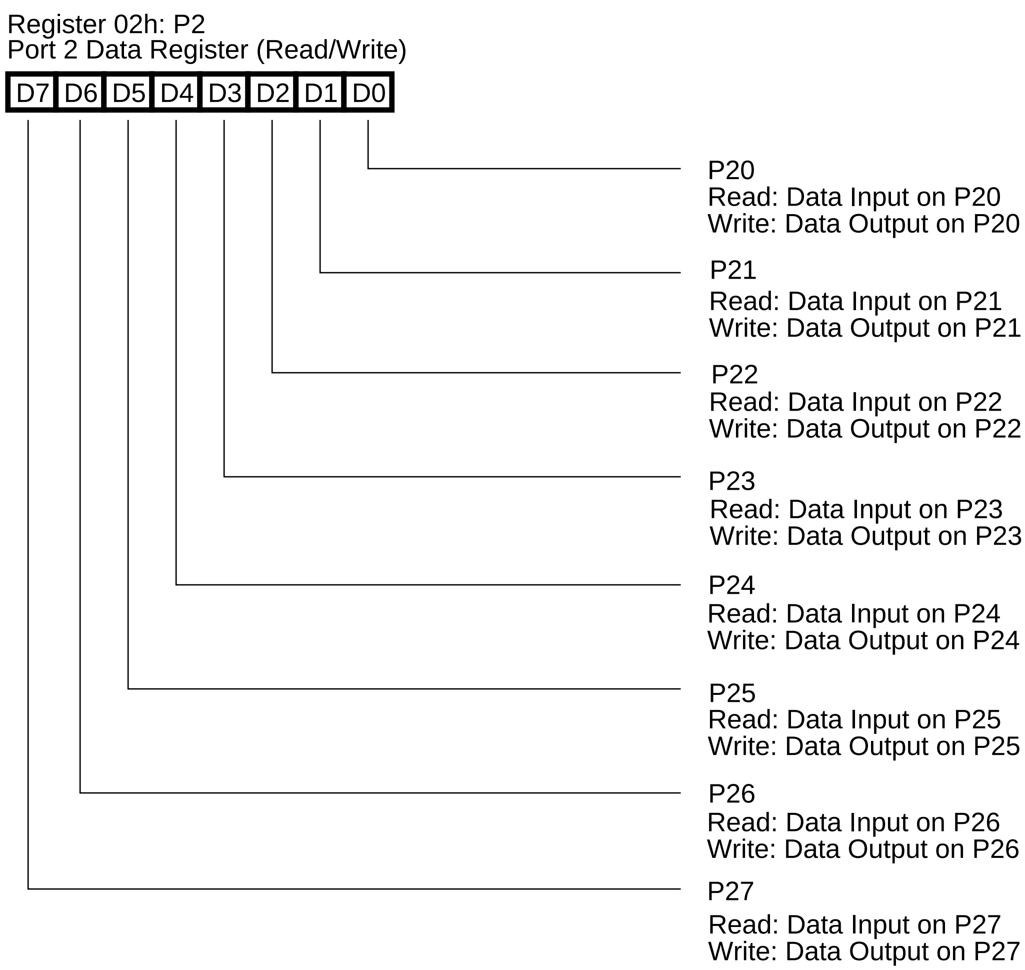


Figure 2-25. Port 2 Data Register

2.4 OPERATING CHARACTERISTICS

Stress outside the levels listed under Operational Limits may cause permanent damage to the device. These limits represent stress limits only, not optimal operating levels. Exposure to maximum rating conditions for extended periods may affect device reliability.

Table 2-5. Operational Limits

Symbol	Parameters	Min	Max	Units	Notes
V_{CC}	Power Supply Voltage	-0.3	+7	V	
V_I	Input Voltage	-0.3	V_{CC}+0.3	V	
V_O	Output Voltage	-0.3	V_{CC}+0.3	V	
I_{OH}	Output Current - High		-10	mA	One pin
I_{OH}	Output Current - High		-100	mA	Total, all pins
I_{OL}	Output Current - Low		20	mA	One pin
I_{OL}	Output Current - Low		200	mA	Total, all pins
T_A	Operating Temperature	0	70	^oC	
T_{STG}	Storage Temperature	-55	150	^oC	

A typical value is 25^{o}C. Minimum and maximum values are 0^{o}C and 70^{o}C respectively.

2.4.1 DC Characteristics

Table 2-6. DC Characteristics

Symbol	Parameter	Min	Typical	Max	Units	Conditions
V_{CC}	Power Supply Voltage	4.5	5.00	5.5	V	
V_{IH}	Input Voltage High	$0.7V_{CC}$		V_{CC}	V	
V_{IL}	Input Voltage Low	0		$0.2V_{CC}$	V	
V_{IHC}	Input XTAL/Oscillator Input High	$0.7V_{CC}$		V_{CC}	V	
V_{ILC}	Input XTAL/Oscillator Input Low	-0.3		$0.2V_{CC}$	V	
V_{OH_ST}	Output Voltage High	V_{CC}-0.4	4.75		V	I_{OH}=-2.00mA
V_{OL_ST}	Output Voltage Low		0.16	0.4	V	I_{OL}=2.00mA
V_{HY}	Schmitt Hysteresis	$0.1V_{CC}$	0.8		V	
I_{IR}	Reset Input Current		-46	-80	uA	V_{RL}=0V
I_{IL}	Input Leakage	-3.0	0.01	3.0	uA	0V, V_{CC}
I_{OL}	Tri-State Leakage	-3.0	0.02	3.0	uA	0V, V_{CC}
I_{CC}	Supply Current		25	40	mA	All inputs at rail; outputs floating
I_{CC1}	Halt Mode Current		9	14	mA	All inputs at rail; outputs floating
I_{CC2}	Stop Mode Current		5	10	uA	All inputs at rail; outputs floating

2.4.2 AC Characteristics

The numbers in Table 2-7 correspond to the numbered signal segments in Figure 2-26.

Table 2-7. AC Characteristics

No.	Symbol	Parameter	Min	Max	Unit
1	TpC	Input Clock Period	166	1000	ns
2	T_RC, T_FC	Clock Input Rise And Fall Time		25	ns
3	T_WC	Input Clock Width	35		ns
4	$T_WHsync_{IN}L$	Hsync Input Low Width	70		ns
5	$T_WHsync_{IN}H$	Hsync Input High Width	3TpC		
6	$TpHsync_{IN}$	Hsync Input Period	8TpC		
7	T_RHsync_{IN}, T_RHsync_{IN}	Hsync Input Rise And Fall Time		100	ns
8	T_WIL	Interrupt Request Input Low	70		ns
9	T_WIH	Interrupt Request Input High	3TpC		
10	T_DPOR	Power-On Reset Delay	25	100	ms
11	$T_DLVIRES$	Low Voltage Detect To Internal Reset Condition	200		ns
12	T_WRES	Reset Minimum Width	5TpC		
13	T_DH_SOI	H_{sync} Start To OSDX2 Stop	2TpV	3TpV	
14	T_DH_SOH	H_{sync} Start To OSDX2 Start		1TpV	

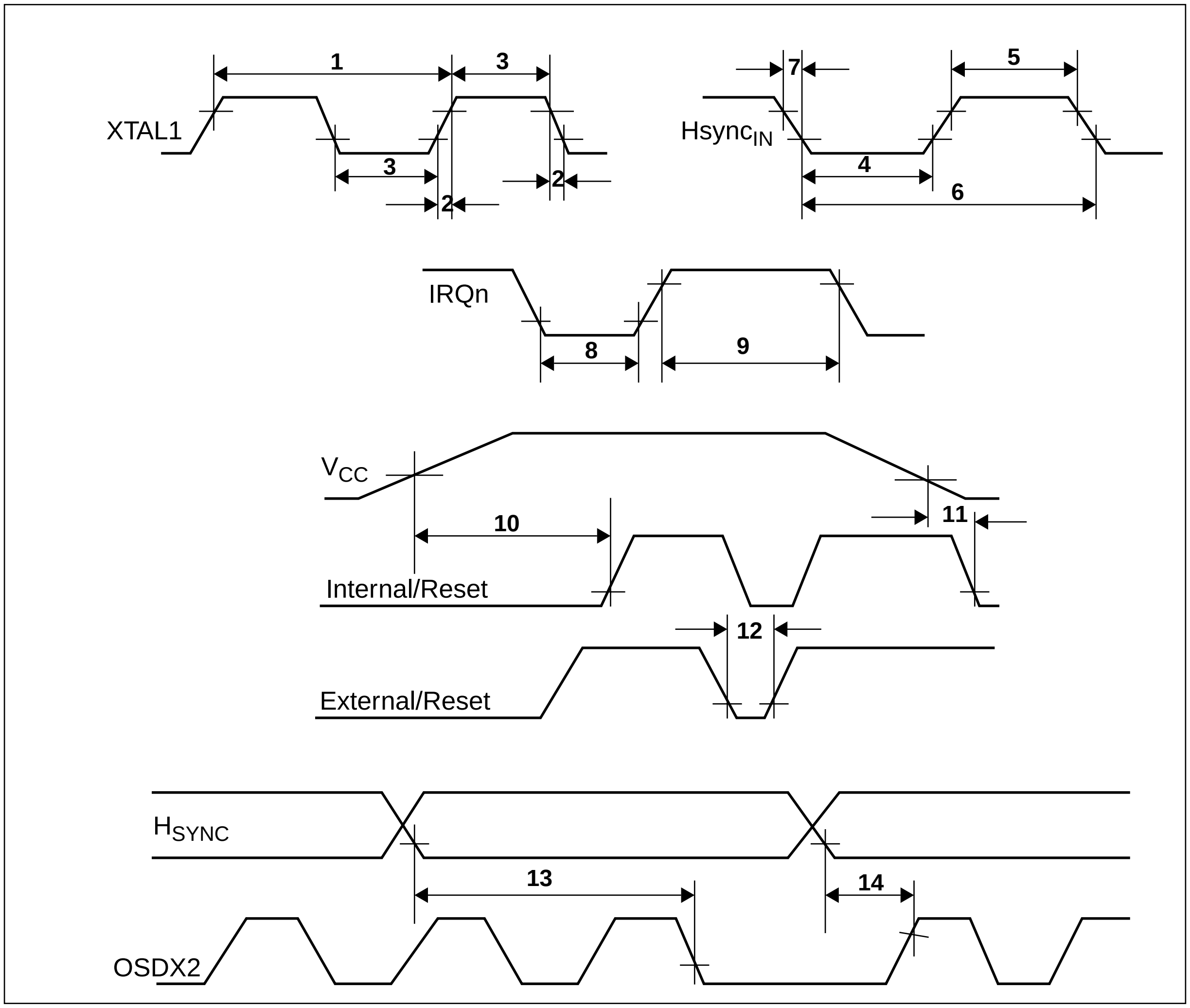


Figure 2-26. AC Characteristics

CHAPTER 3
INTERNAL MICROPROCESSOR OVERVIEW

3.1 ADDRESS SPACE

Four address spaces are available for the Z90200 Family of Digital Television Controllers (DTCs):

- The Standard Register File contains addresses for peripheral, control, all general-purpose, and all I/O port registers. This is the default register file specification.
- The Expanded Register File contains addresses for control and data registers for additional peripherals/features.
- External Program Memory contains addresses for all memory locations having executable code and/or data.
- External Data Memory contains addresses for all memory locations that hold data only.

3.2 STANDARD REGISTER FILE

The Standard Register File consists of up to 256 consecutive bytes (registers). The register file consists of 1 I/O port (02h), 236 General-Purpose Registers (04h-EFh), and 16 Control Registers (F0h-FFh). Registers 00h, 01h, and 03h are reserved. Table 3-1 shows the layout of the register file, including register names, locations, and identifiers.

Table 3-1. Working Register Groups

Register Pointer (FDh) High Nibble	Working Register Group (Hex)	Actual Registers (Hex)
1111(b)	F	F0–FF
1110(b)	E	E0–EF
1101(b)	D	D0–DF
1100(b)	C	C0–CF
1011(b)	B	B0–BF
1010(b)	A	A0–AF
1001(b)	9	90–9F
1000(b)	8	80–8F
0111(b)	7	70–7F
0110(b)	6	60–6F
0101(b)	5	50–5F
0100(b)	4	40–4F
0011(b)	3	30–3F
0010(b)	2	20–2F
0001(b)	1	10–1F
0000(b)	0	00–0F

Registers can be accessed as either 8-bit or 16-bit registers using Direct, Indirect, or Indexed Addressing. All 236 general-purpose registers can be referenced or modified by any instruction that accesses an 8-bit register, without the need for special instructions. Registers accessed as 16 bits are treated as even-odd register pairs (there are 118 valid pairs). In this case, the data's Upper Byte (UB) is stored in the even-numbered register, while the Lower Byte (LB) goes into the next higher odd-numbered register.

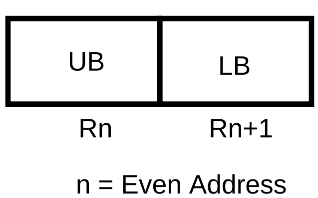


Figure 3-1. 16-Bit Register Addressing

By using a logical instruction and a mask, individual bits within registers can be accessed for bit-set, bit-clear, bit-complement, or bit-test operations. For example, the instruction AND R15, MASK performs a bit-clear operation.

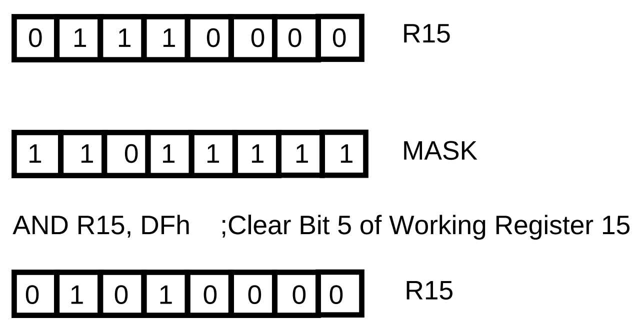


Figure 3-2. Accessing Individual Bits (Example)

When instructions are executed, registers are read when defined as sources and written when defined as destinations. All general-purpose registers function as accumulators, address pointers, index registers, stack areas, or scratch pad memory.

3.2.1 General-Purpose Registers

General-Purpose Registers (GPR) are undefined after the device is powered up. The registers keep their last value after any reset, as long as the reset occurs in the V_{CC} voltage-specified operating range. It does not keep its last state from a V_{LV} reset if V_{CC} drops below 1.8V.

Note: Registers in banks E0-EF may only be accessed through the working register and indirect addressing modes. Direct access cannot be used because the 4-bit working register address mode already uses the format [E| dst], where dst represents the working register number from 0h to Fh.

3.2.2 RAM Protect

The upper portion of the register file address space 80h to EFh (excluding the control registers) may be protected from reading and writing. The RAM Protect bit option is mask-programmable and is selected by the customer when the ROM code is submitted. After the mask option is selected, the user activates this feature from the internal ROM code to turn off/on the RAM protect by loading either a 0 or 1 into IMR (D6). A 1 in D6 enables RAM protect. Only devices that use registers 80h to EFh offer this feature.

3.2.3 Working Register Groups

Instructions can access 8-bit registers and register pairs (16-bit words) using either 4-bit or 8-bit address fields. 8-bit address fields refer to the actual address of the register. For example, register 58h is accessed by calling upon its 8-bit binary equivalent, 01011000 (58h).

With 4-bit addressing, the register file is logically divided into 16 Working Register Groups of 16 registers each, as shown in Table 3-4. These 16 registers are known as Working Registers. A Register Pointer (one of the control registers, FDh) contains the base address of the active working register group. The high nibble of the register pointer determines the current Working Register Group.

When accessing one of the working registers, the 4-bit address of the working register is combined within the upper four bits (high nibble) of the register pointer, forming the actual 8-bit address. Figure 3-3 illustrates this operation. Since working registers are typically specified by short-format instructions, fewer bytes of code are needed, which reduces execution time. In addition, when processing interrupts or changing tasks, the register pointer speeds context switching. A special Set Register Pointer (SRP) instruction assigns a new value to the register pointer.

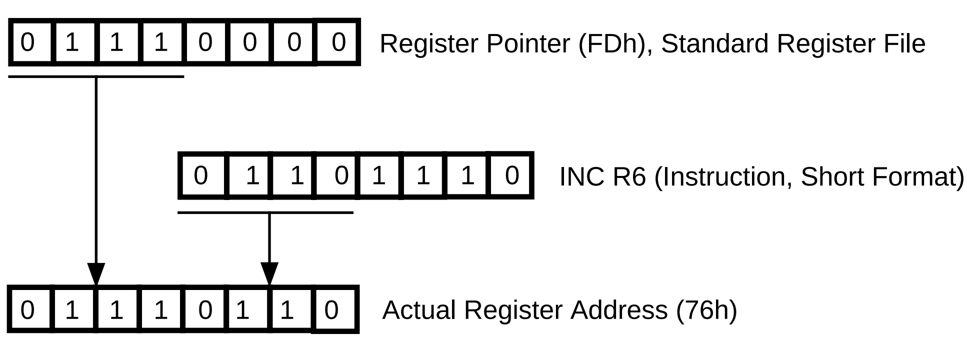


Figure 3-3. Working Register Addressing Examples

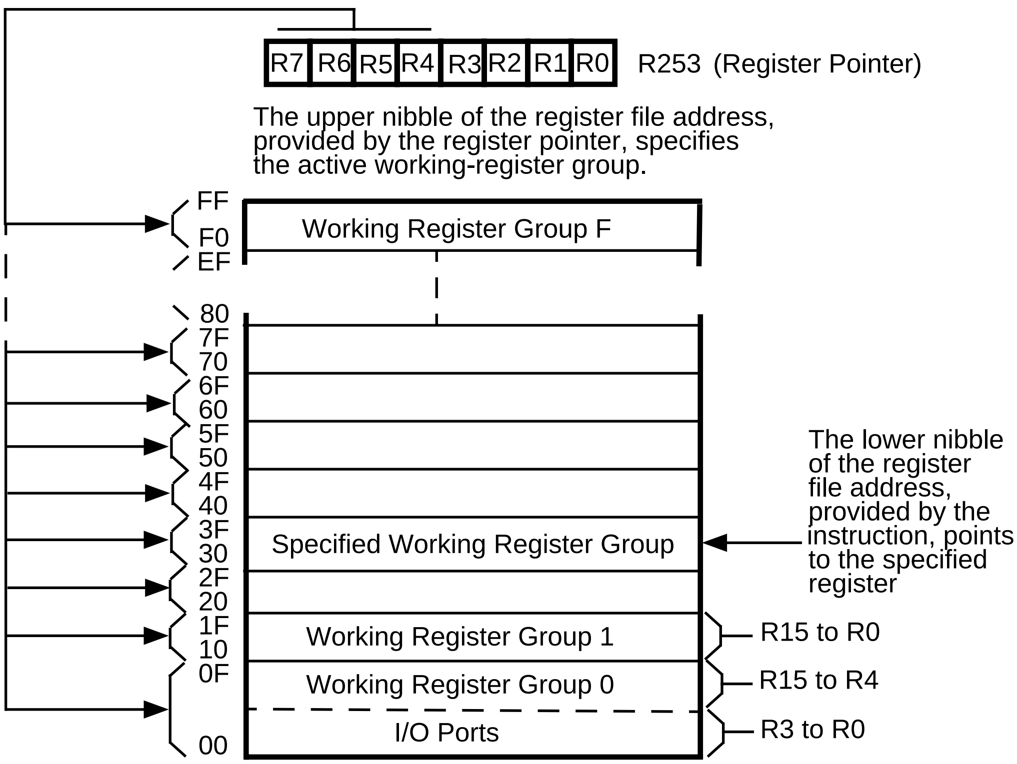


Figure 3-4. Register Pointer

Note:

1. The full register file is shown. Please refer to the selected device product specification for the actual file size.

3.2.4 Error Conditions

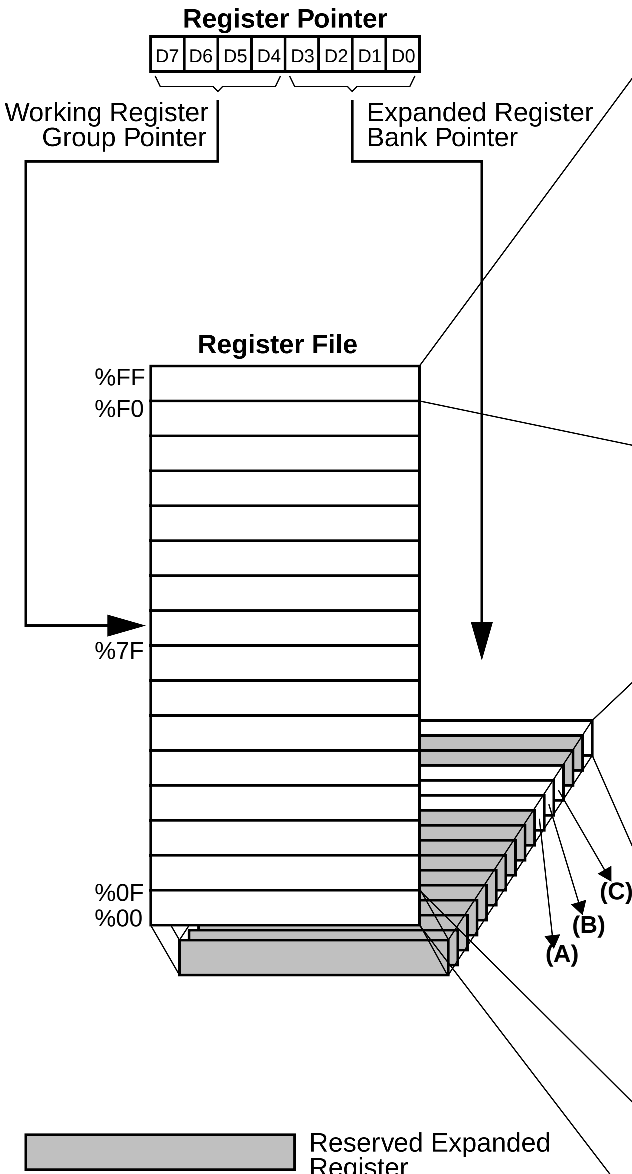
Registers in the Standard Register File must be correctly used because certain conditions produce inconsistent results and should be avoided.

- Registers F3h and F5h-F9h are Write-Only registers. If an attempt is made to read these registers, FFh is returned. Reading any Write-Only register returns FFh.
- When register FDh (register pointer) is read, the least significant four bits (lower nibble) indicate the current Expanded Register File Bank. For example: 0000 indicates the standard register file, while 1010 indicates Expanded Register File Bank A.
- When Ports 0 and 1 are defined as address outputs, registers 00h and 01h return 1s in each address bit location when read.
- Writing bits that are defined as timer output, serial output, or handshake output have no effect.
- The instruction DJNZ uses any general-purpose working register as a counter.
- Logical instructions such as OR and AND require that the current contents of the operand be read. Therefore, they do not function properly on Write-Only registers.

- The WDTMR register must be written within the first 60 internal system clocks cycles of operation after a Reset.

3.3 EXPANDED REGISTER FILE

The standard register file has been expanded to form 16 Expanded Register File (ERF) Banks (Figure 3-5). Each ERF bank consists of up to 256 registers (the same amount as in the standard register file) that can then be divided into 16 working register groups. This expansion allows for access to additional feature/peripheral control and data registers.



Register		D7	D6	D5	D4	D3	D2	D1	D0
%FF	SPL	U	U	U	U	U	U	U	U
%FE	SPH	U	U	U	U	U	U	U	U
%FD	RP	0	0	0	0	0	0	0	0
%FC	FLAGS	U	U	U	U	U	U	U	U
%FB	IMR	0	U	U	U	U	U	U	U
%FA	IRQ	0	0	0	0	0	0	0	0
%F9	IPR	U	U	U	U	U	U	U	U
%F8	P01M	1	1	1	1	1	1	1	1
%F7	P2CNTL	0	0	0	0	0	0	0	1
%F6	P2M	1	1	1	1	1	1	1	1
%F5	PRE0	U	U	U	U	U	U	U	0
%F4	T0	U	U	U	U	U	U	U	U
%F3	PRE1	U	U	U	U	U	U	0	0
%F2	T1	U	U	U	U	U	U	U	U
%F1	TMR	0	0	0	0	0	0	0	0
%F0	Reserved								

Reset Condition

Expanded Register Bank (F)

Register		D7	D6	D5	D4	D3	D2	D1	D0
%(F)0F	WDTMR	U	U	U	0	1	1	0	1
%(F)0E	Reserved								
%(F)0D	Reserved								
%(F)0C	Reserved								
%(F)0B	SMR	0	0	1	0	0	0	0	0
%(F)0A	Reserved								
%(F)09	Reserved								
%(F)08	Reserved								
%(F)07	MC_Reg	0	0	0	0	0	0	0	0
%(F)06	MR_En	0	0	0	0	0	0	0	0
%(F)05	MC_End	U	U	U	0	0	0	0	0
%(F)04	MC_St	U	U	U	0	0	0	0	0
%(F)03	PRT6_DTA	U	U	U	U	1	1	1	1
%(F)02	PRT6_DRT	1	1	1	1	1	1	1	1
%(F)01	4ADC_DTA	0	0	0	0	U	U	U	U
%(F)00	PCON	1	1	1	1	1	1	1	0

Reset Condition

Register		D7	D6	D5	D4	D3	D2	D1	D0
%(0)03	Reserved								
%(0)02	P2	U	U	U	U	U	U	U	U
%(0)01	Reserved								
%(0)00	Reserved								

Figure 3-5. Register and Expanded Register File Map

Currently, 4 of the 16 possible ERF banks have been implemented. ERF bank 0, also known as the Standard Register File, has all 256 bytes defined (Figure 3-5). Only working register group 0 (register addresses 00h to 0Fh) have been defined for ERF bank C and ERF bank F. All other working register groups in ERF banks C and F, as well as the remaining 13 ERF banks, are unimplemented. All are reserved for future use.

When an ERF Bank is selected, register addresses 00h to 0Fh access those sixteen ERF bank registers—in effect replacing the first sixteen locations of the standard register file.

For example, if ERF bank C is selected, the standard registers 00h through 0Fh are no longer accessible. Registers 00h through 0Fh are now the 16 registers from ERF bank C, working register group 0. No other standard registers are effected since only working register group 0 is implemented in ERF bank C.

Access to the ERF is accomplished through the register pointer (FDh). The lower nibble of the register pointer determines the ERF bank while the upper nibble determines the working register group within the register file.

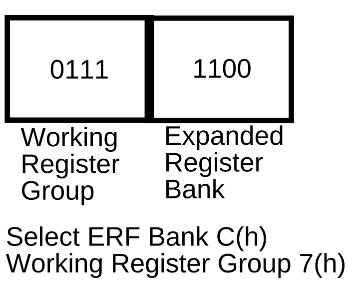


Figure 3-6. Register Pointer (FDh) Example

The value of the lower nibble in the register pointer (FDh) corresponds to the ERF bank identification. Table 3-2 shows the lower nibble value and the register file assigned to it.

The upper nibble of the register pointer selects the group of 16 bytes in the register file, out of the full 256, to be accessed as working registers.

R253 RP = 00h	;ERF Bank 0, Working Reg. Group 0.	
	R0 = Reserved = 00h	
	R1 = Reserved = 01h	
	R2 = Port 2 = 02h	
	R3 = Reserved = 03h	
	R11 = GPR 0Bh	
	R15 = GPR 0Fh	
If:		
R253 RP = 0Fh	;ERF Bank F, Working Reg. Group 0.	
	R0 = PCON = 00h	
	R1 = 4ADC_DTA = 01h	
	R2 = PRT6_DRT= 02h	
	R11 = SMR = 0Bh	
	R15 = WDTMR = 0Fh	
If:		
R253 RP = FFh	;ERF Bank F, Working Reg. Group F.	
	R0 = Reserved	00h= PCON
	R1 = TMR	01h= 4ADC_DTA
	R2 = T1	02h= PRT6_DRT
		...
	R11 = IMR	0Bh = SMR
		...
	R15 = SPL	0Fh = WDTMR

Note: Enabling an ERF bank (C or F) only changes register addresses 00h to 0Fh; the working register pointer can be used to access either the selected ERF bank (bank C or F, working register group 0) or the Standard Register File (ERF bank 0, working register groups 1 through F).

Note: When an ERF bank other than bank 0 is enabled, the first 16 bytes of the standard register file (I/O ports 0 to 3, Groups 4 to F) are no longer accessible. The selected ERF bank, registers 00h to 0Fh are accessed instead. It is important to re-initialize the register pointer to enable ERF bank 0 when these registers are required for use.

3.4 CONTROL AND PERIPHERAL REGISTERS

3.4.1 Standard Registers

The standard control registers govern the operation of the CPU. Any instruction which references the register file can access these control registers. Available control registers are:

- Interrupt Priority Register (IPR)
- Interrupt Mask Register (IMR)
- Interrupt Request Register (IRQ)
- Program Control Flags (FLAGS)
- Register Pointer (RP)
- Stack Pointer Upper Byte (SPH)
- Stack Pointer Lower Byte (SPL)

A 16-bit Program Counter (PC) determines the sequence of current program instructions. The PC is not an addressable register.

Peripheral registers are used to transfer data, configure the operating mode, and control the operation of the on-chip peripherals. Any instruction that references the register file can access the peripheral registers. The peripheral registers are:

- Timer Mode (TMR)
- Timer/Counter 0 (T0)
- T0 Prescaler (PRE0)
- Timer/Counter 1 (T1)
- T1 Prescaler (PRE 1)
- Port 2 Mode (P2M)
- Port 2 Output Control (P2CNTL)

In addition, the port register (P2) is considered to be a peripheral register.

3.4.2 Expanded Registers

The expanded control registers govern the operation of additional features or peripherals. Any instruction which references the register file can access these registers.

Working register group 0 in ERF bank A consists of the registers for the On-Screen Display (OSD). Table 3-2 shows the registers within this group.

Table 3-2. Expanded Register File Bank A

Register	Register Function	Working Register
F	CLR_P6	R15
E	CLR_P5	R14
D	CLR_P4	R13
C	CLR_P3	R12
B	CLR_P2	R11
A	CLR_P1	R10
9	CLR_P0	R9

Table 3-2. Expanded Register File Bank A

Register	Register Function	Working Register
8	SNDCLR	R8
7	SNDCLR_CNTRL	R7
6	FADE_POS	R6
5	FADE_POS	R5
4	ROW_SPACE	R4
3	DISP_ATTR	R3
2	HOR_POS	R2
1	VERT_POS	R1
0	OSD_CNTL	R0

Working register group 0 in ERF bank B consists of the registers for the pulse-width modulators. Table 3-3 shows the registers within this group.

Table 3-3. Expanded Register File Bank B

Register	Register Function	Working Register
E	PRT5_DRT	R14
D	P_MODE	R13
C	PRT5_DTA	R12
B	PWM10	R11
A	PWM9	R10
9	PWM8	R9
8	PWM7	R8
7	PWM6	R7
6	PWM5	R6
5	PWM4	R5
4	PWM3	R4
3	PWM2	R3
2	PWM1	R2
1	PWM11L	R1
0	PWM11H	R0

Register bank C in the ERF consists of the registers for the I^2C interface. Table 3-4 shows the registers within ERF bank C, working register group 0.

Table 3-4. Expanded Register File Bank C

Register	Register Function	Working Register
C	I^2C_CNTL	R12
B	I^2C_CMD	R11
A	I^2C_DATA	R10
9	CLR_IDX	R9
8	PIN_SLT	R8
7	INT_ST	R7
6	PRT4_DRT	R6
5	PRT4_DTA	R5
4	IR_CP1	R4
3	IR_CP0	R3
2	TCR1	R2
1	TCR0	R1
0	3ADC_DTA	R0

Working register group 0 in ERF bank F consists of the control registers for Stop mode, WDT, and port control. Figure 3-5 shows the registers within this group.

Table 3-5. Expanded Register File Bank F

Register	Register Function	Working Register
F	WDTMR	R15
E	Reserved	R14
D	Reserved	R13
C	Reserved	R12
B	SMR	R11
A	Reserved	R10
9	Reserved	R9
8	Reserved	R8
7	Mesh Control Register (MC_Reg)	R7
6	Mesh Row Enable (MR_En)	R6
5	Mesh Column End (MC_End)	R5
4	Mesh Column Start (MC_St)	R4
3	PRT6_DTA	R3
2	PRT6_DRT	R2
1	4ADC_DTA	R1
0	PCON	R0

The functions and applications of the control and peripheral registers are described in other sections of this manual.

3.5 PROGRAM MEMORY

The first 12 bytes of Program Memory are reserved for the interrupt vectors. These locations contain six 16-bit vectors that correspond to the six available interrupts. Address 12 up to the maximum ROM address consists of on-chip mask-programmable ROM. See the product data sheet for the exact program, data, register memory size, and address range available. The internal program memory is one-time programmable (OTP) or mask programmable dependent on the specific device.

Note: A ROM protect feature prevents the dumping of ROM contents by inhibiting execution of the LDC, LDCI, LDE, and LDEI instructions to program memory in all modes.

The ROM Protect option is mask-programmable,

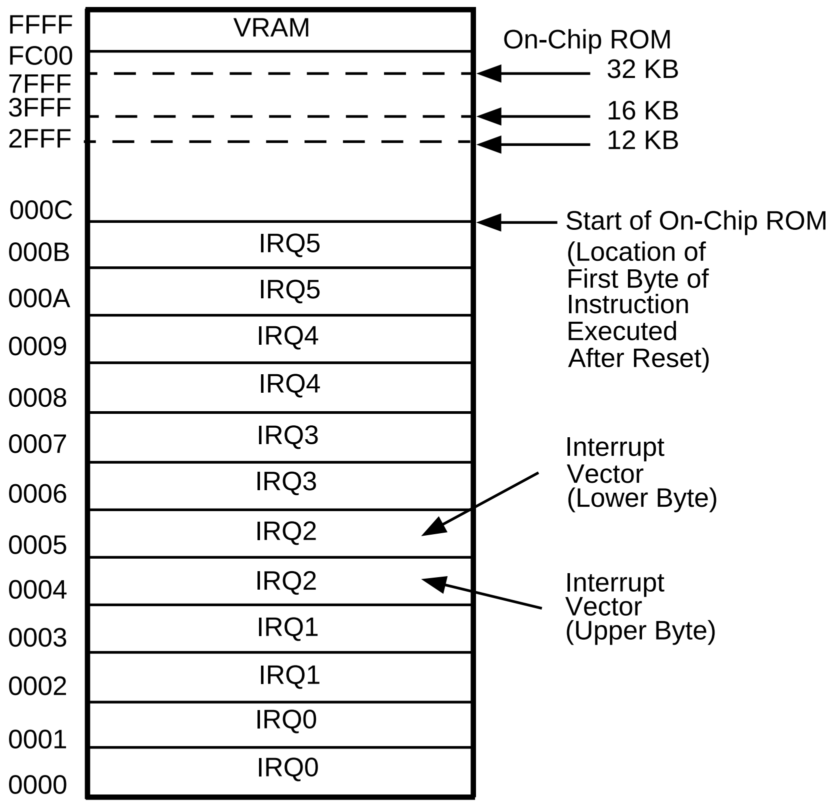


Figure 3-7. Program Memory Map

3.6 STACKS

The register pair FEh and FFh form the 16-bit Stack Pointer (SP), that is used for all stack operations. The stack address is stored with the UB in FEh and LB in FFh.

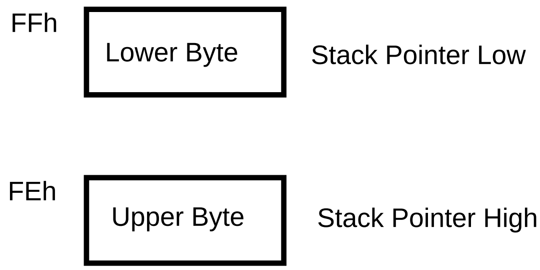


Figure 3-8. Stack Pointer

The stack address is decremented prior to a PUSH operation and incremented after a POP operation. The stack address always points to the data stored on the top of the stack. The stack is a return stack for CALL instructions and interrupts, as well as a data stack.

During a CALL instruction, the contents of the PC are saved on the stack. The PC is restored during a RETURN instruction. Interrupts cause the contents of the PC and Flag registers to be saved on the stack. The IRET instruction restores them (Figure 3-9).

When the microcontroller is configured for an internal stack (using the Standard Register File), register FFh serves as the Stack Pointer. The value in FEh is ignored. FEh can be used as a general-purpose register in this case only.

An overflow or underflow can occur when the stack address is incremented or decremented during normal stack operations. If not prevented, an unpredictable operation occurs.

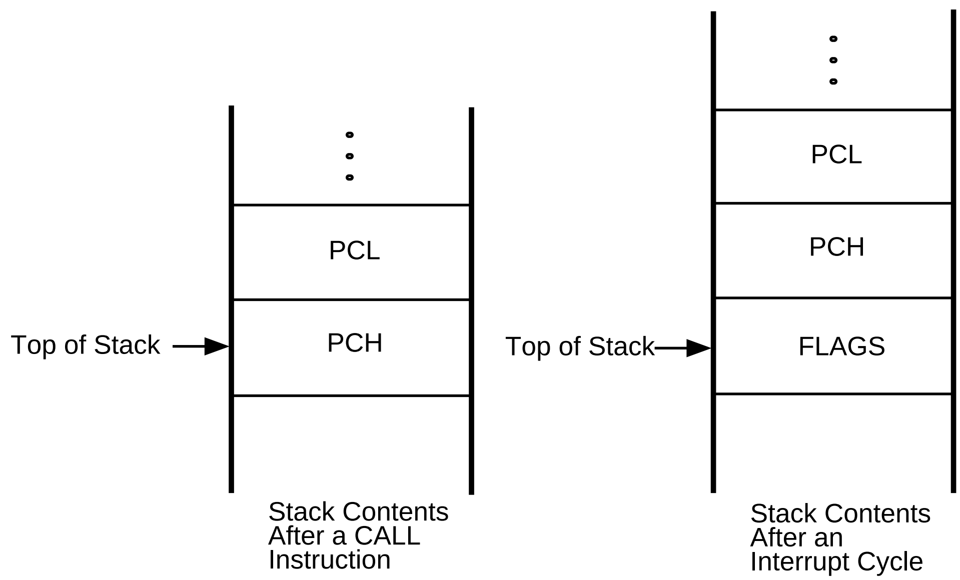


Figure 3-9. Stack Operations

3.7 OSCILLATOR CONTROL

In some cases, the microcontroller offers software control of the oscillator to select low EMI drive or standard drive. The selection is done by programming bit D7 of the Port Configuration (PCON) register. The PCON register is located in Expanded register file bank F, register 00h.

A 1 in bit D7 configures the oscillator with standard drive, while a 0 configures the oscillator with Low EMI drive. This only affects the drive capability of the oscillator and does not affect the relationship of the XTAL clock frequency to the internal system clock (SCLK).

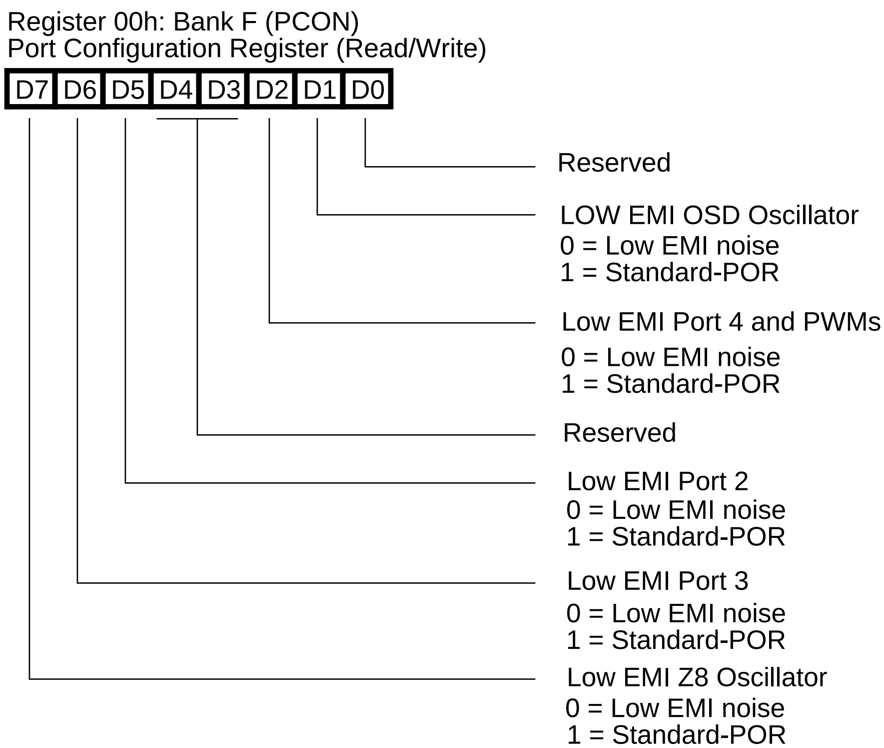


Figure 3-10. Port Configuration Register

3.8 OSCILLATOR OPERATION

The microcontroller uses a Pierce oscillator with an internal feedback. The advantages of this circuit are low cost, large output signal, low-power level in the crystal, stability with respect to V_{CC} and temperature, and low impedances (not disturbed by stray effects).

One draw back is the need for high gain in the amplifier to compensate for feedback path losses. The oscillator amplifies its own noise at start-up until it settles at the frequency that satisfies the gain/phase requirements A x B = 1, where A = V0/Vi is the gain of the amplifier and B = Vi/V0 is the gain of the feedback element. The total phase shift around the loop is forced to zero (360 degrees). Since V_{IN} must be in phase with itself, the amplifier/inverter provides 180 degree phase shift and the feedback element is forced to provide the other 180 degrees of phase shift.

R1 is a resistive component placed from output to input of the amplifier. The purpose of this feedback is to bias the amplifier in its linear region and to provide the start-up transition.

Capacitor C2 combined with the amplifier output resistance provides a small phase shift. It also provides some attenuation of overtones.

Capacitor C1 combined with the crystal resistance provides additional phase shift.

C1 and C2 can affect the start-up time if they increase dramatically in size. As C1 and C2 increase, the start-up time increases until the oscillator reaches a point where it does not start up any more.

It is recommended for fast and reliable oscillator start-up (over the manufacturing process range) that the load capacitors be sized as low as possible without resulting in overtone operation.

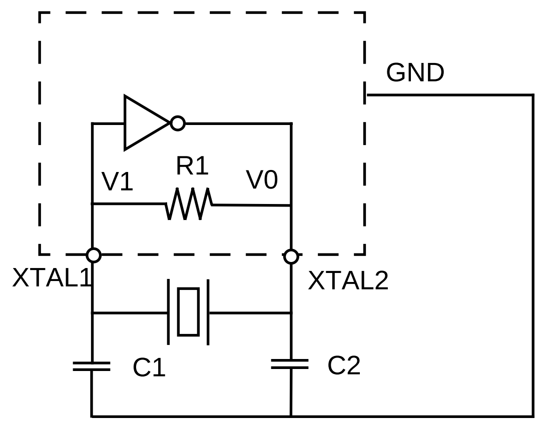


Figure 3-11. Pierce Oscillator with Internal Feedback Circuit

3.8.1 Layout

Traces connecting crystal, caps, and the oscillator pins should be as short and wide as possible. This reduces parasitic inductance and resistance. The components (caps, crystal, resistors) should be placed as close as possible to the oscillator pins.

The traces from the oscillator pins of the IC and the ground side of the lead caps should be guarded from all other traces (clock, V_{CC}, address/data lines, system ground) to reduce cross talk and noise injection. This is usually accomplished by keeping other traces and system ground trace planes away from the oscillator circuit and by placing a device V_{SS} ground ring around the traces/components. The ground side of the oscillator lead caps should be connected to a single trace to the GND pin. It should not be shared with any other system ground trace or components except at the GND pin. This is to prevent differential system ground noise injection into the oscillator (Figure 3-11).

3.8.2 Indications of an Unreliable Design

There are two major indicators that are used in working designs to determine their reliability over full lot and temperature variations. They are:

- **Start-Up Time:** If start-up time is excessive, or varies widely from unit to unit, there is probably a gain problem. C1/C2 should be reduced; the amplifier gain is not adequate at frequency, or crystal Rs is too large.

- **Output Level:** The signal at the amplifier output should swing from ground to V_{CC}. This indicates there is adequate gain in the amplifier. As the oscillator starts up, the signal amplitude grows until clipping occurs, at which point the loop gain is effectively reduced to unity and constant oscillation is achieved. A signal of less than 2.5 volts peak-to-peak is an indication that low gain may be a problem. Either C1 or C2 should be made smaller or a low-resistance crystal should be used.

3.8.3 Circuit Board Design Rules

The following circuit board design rules are suggested:

- To prevent induced noise the crystal and load capacitors should be physically located as close to the microcontroller as possible.
- Signal lines should not run parallel to the clock oscillator inputs. In particular, the crystal input circuitry and the internal system clock output should be separated as much as possible.
- V_{CC} power lines should be separated from the clock oscillator input circuitry.
- Resistivity between XTAL1 or XTAL2 and the other pins should be greater than 10 Mohms.

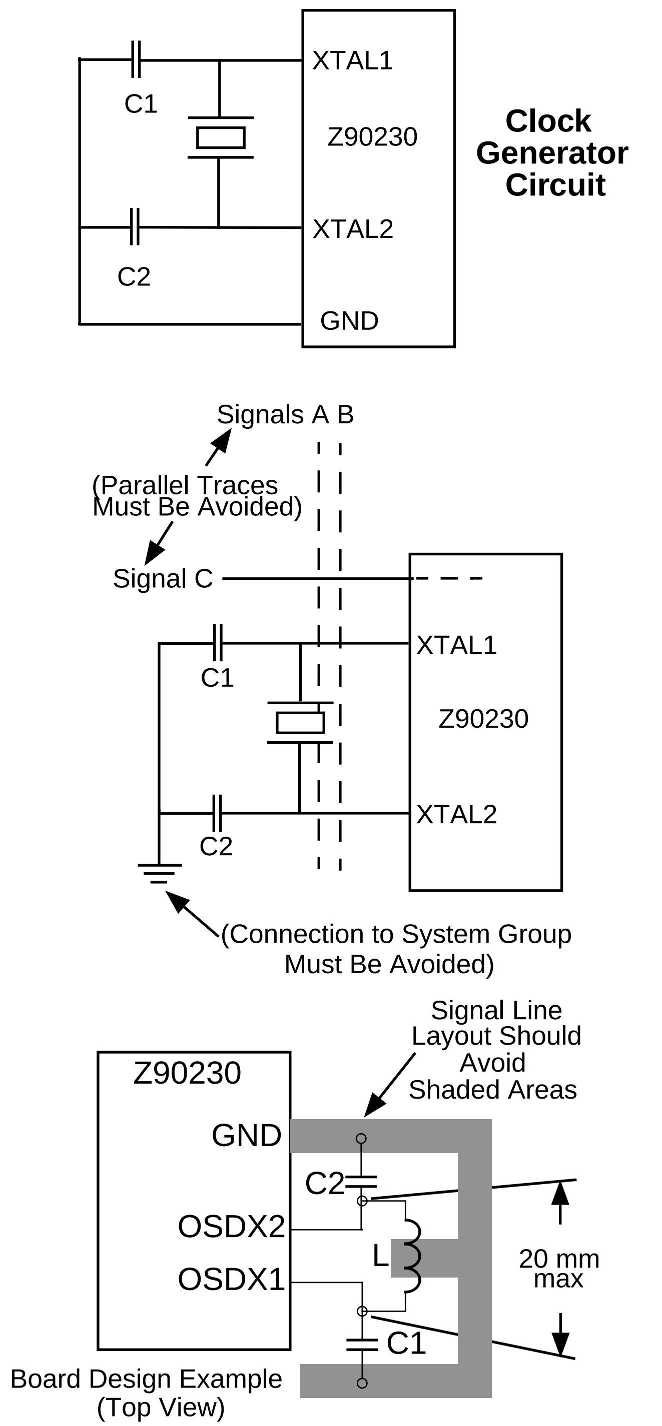


Figure 3-12. Circuit Board Design Rules

3.8.4 Crystals and Resonators

Crystals and ceramic resonators should have the following characteristics to ensure proper oscillator operation:

Crystal cut	AT (crystal only)
Mode	Parallel, Fundamental Mode
Crystal capacitance	<7pF
Load capacitance	10pF < CL < 220 pF,
	15 typical
Resistance	100 ohms max

Depending on the operation frequency, the oscillator may require the addition of capacitors C1 and C2 (shown in Figure 3-13).

The capacitance values are dependent on the manufacturer's crystal specifications.

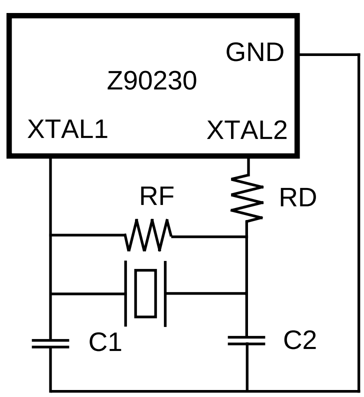


Figure 3-13. Crystal/Ceramic Resonator Oscillator

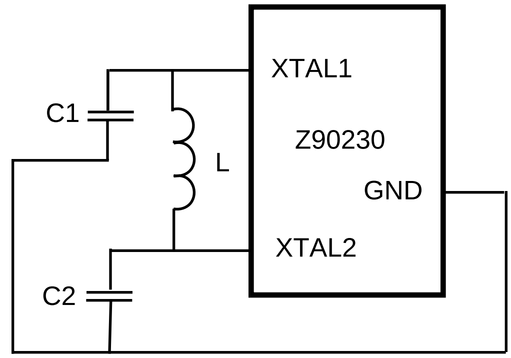


Figure 3-14. LC Clock

In most cases, the RD is 0 Ohms and RF is infinite. It is determined and specified by the crystal/ceramic resonator manufacturer. The RD can be increased to decrease the amount of drive from the oscillator output to the crystal. It can also be used as an adjustment to avoid clipping of the oscillator signal to reduce noise. The RF can be used to improve the start-up of the crystal/ceramic resonator. The oscillator already has an internal shunt resistor in parallel to the crystal/ceramic resonator.

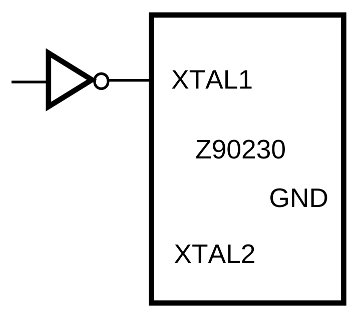


Figure 3-15. External Clock

It is recommended that the load capacitor ground trace be directly connected to the GND pin. This ensures that no system noise is injected into the MCU clock. This trace should not be shared with any other components except at the GND pin.

In some cases, the XTAL1 pin also functions as one of the EPROM high-voltage mode programming pins or as a special factory test pin. In this case, applying 2V above V_{CC} on the XTAL1 pin causes the device to enter one of these modes. Since this pin accepts high voltages to enter these respective modes, the standard input protection diode to V_{CC} is not on XTAL1. It is recommended that in applications where the microcontroller is exposed to high system noise, a diode from XTAL1 to V_{CC} be used to prevent accidental enabling of these modes. This diode does not affect the crystal/ceramic resonator operation. Parallel resonant crystal or resonator data sheets specify a load capacitor value that is the series combination of C_1 and C_2, including all parasitics (PCB and holder).

3.9 LC OSCILLATOR

The oscillator can use a LC network to generate a XTAL clock.

The frequency stays stable over V_{CC} and temperature. The oscillation frequency is determined by the equation:

$$\text{Frequency} = \frac{1}{2\pi\sqrt{LC_T}}$$

where L is the total inductance including parasitics and C_T is the total series capacitance including the parasitics.

Simple series capacitance is calculated using the following equation:

$$\frac{1}{C_T} = \frac{1}{C_1} + \frac{1}{C_2}$$

If $C_1 = C_2$

$$\frac{1}{C_T} = \frac{2}{C_1}$$

$$C_1 = 2C_T$$

Figure 3-16. Capacitance Calculation

Sample calculation of capacitance C_1 and C_2 for 5.83 MHz frequency and inductance value of 27 µH:

$$5.83 \times 10^6 = \frac{1}{2\pi\sqrt{2.7 \times 10^{-6} C_T}}$$

C_T = 27.6 pF

Thus, C_1 = 55.2 pF and C_2 = 55.2 pF.

3.10 RESET—WATCH-DOG TIMER

This section describes the microcontroller reset conditions, reset timing, and register initialization procedures. Reset is generated by Power-On Reset (POR), Reset Pin, Watch-Dog Timer (WDT), and Stop-Mode Recovery.

A system reset overrides all other operating conditions and puts the microcontroller into a known state. To initialize the chip's internal logic, the $\overline{\text{Reset}}$ input must be held Low for at least 5 XTAL clock cycles. The control register and ports are reset to their default conditions after a POR, a reset from the $\overline{\text{Reset}}$ pin, or WDT timeout while in RUN Mode and Halt Mode. The control registers and ports are not reset to their default conditions after Stop-Mode Recovery and WDT timeout while in Stop Mode.

The program counter is loaded with 000Ch. I/O ports and control registers are configured to their default reset state.

Resetting the microcontroller does not effect the contents of the general-purpose registers.

3.11 RESET PIN, INTERNAL POR OPERATION

In some cases, the microcontroller hardware Reset pin initializes the control and peripheral registers. Specific reset values are shown by 1 or 0, while bits whose states are unknown are indicated by the letter U.

Table 3-6. Sample Control and Peripheral Register Reset Values

Register (HEX)	Register Name	Bits 7	6	5	4	3	2	1	0	Comments
F0	Serial I/O	U	U	U	U	U	U	U	U	
F1	Timer Mode	0	0	0	0	0	0	0	0	Counter/Timers Stopped
F2	Counter/Timer1	U	U	U	U	U	U	U	U	
F3	T1 Prescaler	U	U	U	U	U	U	0	0	Single-Pass Count Mode, External Clock Source
F4	Counter/Timer0	U	U	U	U	U	U	U	U	
F5	T0 Prescaler	U	U	U	U	U	U	U	0	Single-Pass Count Mode
F6	Port 2 Mode	1	1	1	1	1	1	1	1	All Inputs
F7	P2CNTL	0	0	0	0	0	0	0	1	Port 2 Open-Drain
F8	Port 0–1 Mode	0	1	0	0	1	1	0	1	Internal Stack, Normal Memory Timing
F9	Interrupt Priority	U	U	U	U	U	U	U	U	
FA	Interrupt Request	0	0	0	0	0	0	0	0	All Interrupts Cleared
FB	Interrupt Mask	0	U	U	U	U	U	U	U	Interrupts Disabled
FC	Flags	U	U	U	U	U	U	U	U	
FD	Register Pointer	0	0	0	0	0	0	0	0	
FE	Stack Pointer (High)	U	U	U	U	U	U	U	U	
FF	Stack Pointer (Low)	U	U	U	U	U	U	U	U	

Program execution starts 5 to 10 clock cycles after Internal Reset has returned High. The initial instruction fetch is from location 000Ch. Figure 3-17 shows Reset timing.

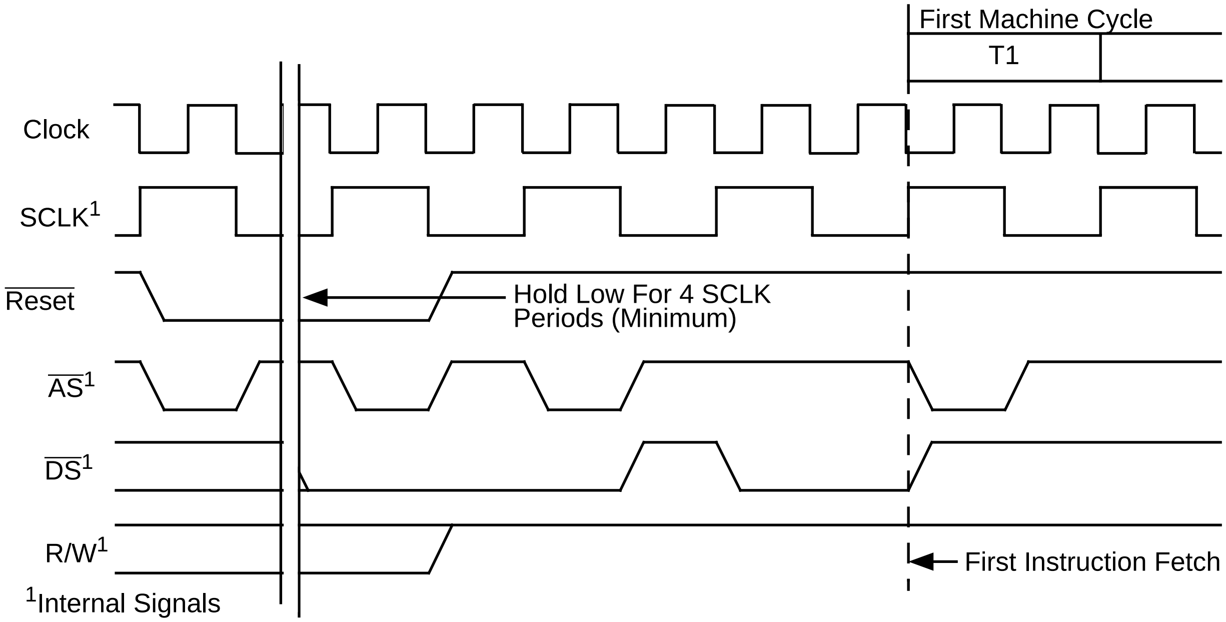


Figure 3-17. Reset Timing

After a reset, the first routine executed should be one that initializes the control registers to the required system configuration.

The Reset pin is the input of a Schmitt-triggered circuit. Resetting the microcontroller initializes port and control registers to their default states. To form the internal reset line, the output of the trigger is synchronized with the internal clock. The clock must therefore be running for Reset to function. It requires four internal system clocks after Reset is detected for the microcontroller to reset the internal circuitry. An internal pull-up, combined with an external capacitor of 1µf, provides enough time to properly reset the microcontroller. The internal POR timer circuit holds the microcontroller in Reset Mode for a duration (T_{POR}) before releasing the device out of reset. The internally generated reset drives the reset pin low for the POR time. Any devices driving the reset line must be open-drained in order to avoid damage from possible conflict during reset conditions. This T_{POR} time allows the on-board clock oscillator to stabilize.

To avoid asynchronous and noisy reset problems, the microcontroller is equipped with a reset filter of four external clocks (4 TpC). If the external reset signal is less than 4 TpC in duration, no reset occurs. On the fifth clock after the Reset is detected, an internal Reset signal is latched and held for an internal register count of 18 external clock cycles, or for the duration of the external Reset, whichever is longer. Program execution begins at location 000Ch, 5-10 TpC cycles after Reset is released. For the internal Power-On Reset, the reset output time is specified as T_{POR}. Please refer to the AC characteristics for actual values.

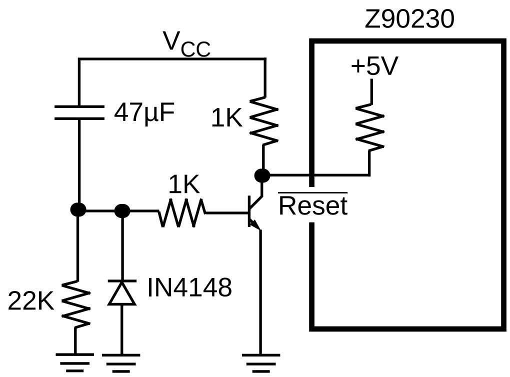


Figure 3-18. External Power-On Reset Circuit Example

Table 3-7. Expanded Register File Bank 0 Reset Values at Reset

Register (HEX)	Register Name	Bits 7	6	5	4	3	2	1	0	Comments
00										N/A
01										N/A
02	Port 2	U	U	U	U	U	U	U	U	Input mode
03										N/A
04–EF	General-Purpose Registers, 04-EF	U	U	U	U	U	U	U	U	Undefined

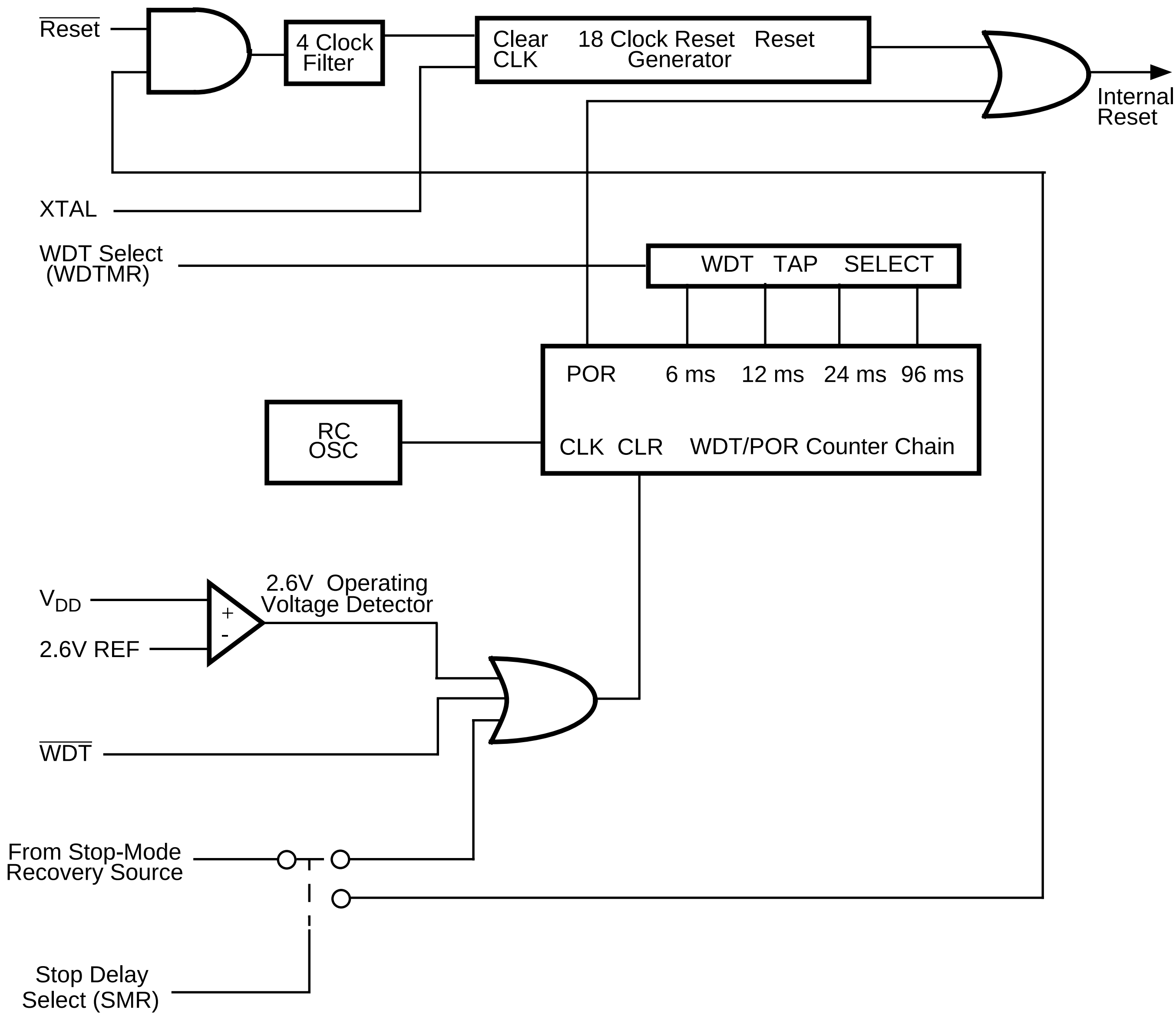


Figure 3-19. Microprocessor Reset with Reset Pin, WDT, SMR, and POR (Example)

3.12 WATCH-DOG TIMER

The Watch-Dog Timer (WDT) is a retriggerable one-shot timer that resets the microcontroller if it reaches its terminal count. When operating in the RUN or Halt Modes, a WDT reset is functionally equivalent to a hardware $\overline{POR}$ reset. The WDT is initially enabled by executing the WDT instruction and refreshed on subsequent executions of the WDT instruction. The WDT cannot be disabled after it has been initially enabled. The WDT can be permanently enabled through a ROM option. Permanently enabled WDTs are always enabled and the WDT instruction is used to refresh it. The WDT circuit is driven by an on-board RC oscillator.

Note: Execution of the WDT instruction affects the Z (zero), S (sign), and V (overflow) flags.

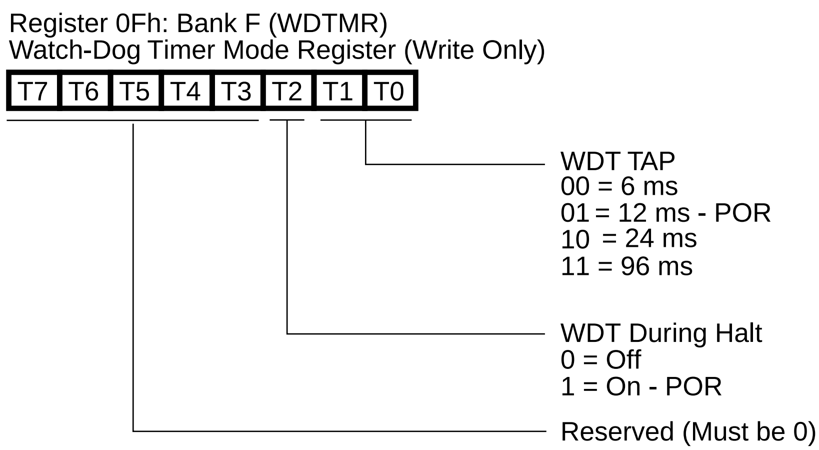


Figure 3-20. Watch-Dog Timer Mode Register (Write-Only) Example

Note: The WDTMR register is accessible only during the first 60 processor cycles from the execution of the first instruction after Power-On Reset, Watch-Dog Reset, or a Stop-Mode Recovery. After this point, the register cannot be modified by any means, intentional or otherwise. The WDTMR is a Write-Only register.

The WDTMR is located in Expanded Register File Bank F, register 0Fh. The control bits are described as follows:

WDT Time Select (T1, T0): Bits 0 and 1 control a tap circuit that determines the time-out period. Table 3-8 shows the different values that can be obtained. The default value of D1 and D0 are 0 and 1, respectively.

Table 3-8. Time-Out Period of the WDT

Time-Out of D1	D0	Minimum Time-Out of Internal RC OSC
0	0	6 ms min
0	1	12 ms min
1	0	24 ms min
1	1	96 ms min

Notes:
The default on reset is, D0 = 1 and D1 = 0.
The values given are for V_{CC} = 5.0V.
See the device product specification for exact WDTMR timeout select options available.

WDT During Halt Mode (T2): Bit 2 determines if the WDT is active during Halt Mode. A 1 value indicates active during Halt. The default is 1. A WDT timeout during Halt Mode resets control register ports to their default reset conditions.

Bits 3, 4, 5, 6 and 7: These bits are reserved.

V_{CC} Voltage Comparator: An on-board voltage comparator checks that V_{CC} is at the required level to ensure correct operation of the device. Reset is globally driven if V_{CC} is below the specified voltage.

3.13 POWER-ON RESET

A timer circuit clocked by a dedicated on-board RC oscillator is used for the Power-On Reset (POR) timer (T_{POR}) function. The POR time allows V_{CC} and the oscillator circuit to stabilize before instruction execution begins.

The POR timer circuit is a one-shot timer triggered by one of three conditions:

1. Power fail to Power OK status (cold start).
2. Stop-Mode Recovery (if bit 5 of SMR=1).
3. WDT timeout.

The POR time is specified as T_{POR}. On the Stop-Mode Recovery register (SMR), bit 5 selects whether the POR timer is used after Stop-Mode Recovery or by-passed. If bit D5 = 1 then the POR timer is used. If bit 5 = 0 then the POR timer is by-passed. In this case, the Stop-Mode Recovery source must be held in the recovery state for 5 TpC or 5 crystal clocks to pass the reset signal internally. This option is used when the clock is provided with an LC clock or an external clock since these clock resources do not require a long stabilization time.

POR always resets the control and port registers to their default condition. In the SMR register, the warm start bit resets to 0 to indicate POR.

3.14 COUNTER/TIMERS

The microcontroller provides up to two 8-bit counter/timers, T0 and T1, each driven by its own 6-bit prescaler, PRE 0 and PRE 1. Both counter/timers are independent of the processor instruction sequence, that relieves software from time-critical operations such as interval timing or event counting. Some MCUs offer clock scaling using the SMR register. The following description is typical.

Each counter/timer operates in either Single-Pass or Continuous Mode. At the end of count, counting either stops or the initial value is reloaded and counting continues. Under software control, new values are loaded immediately or when the end-of-count is reached. Software also controls the counting mode, how a counter/timer is started or stopped, and its use of I/0 lines. Both the counter and prescaler registers can be altered while the counter/timer is running.

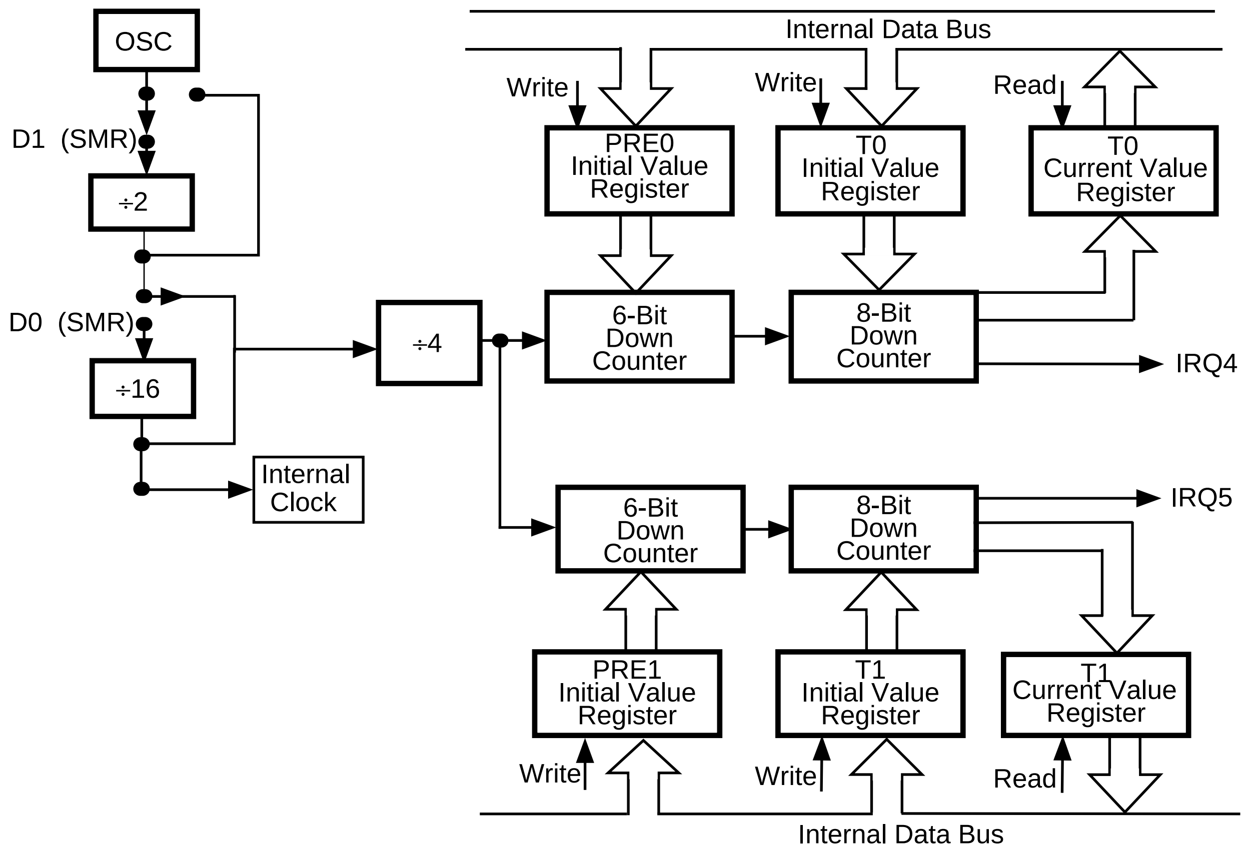


Figure 3-21. Counter/Timers Block Diagram

Counter/Timers 0 and 1 are driven by a timer clock generated by dividing the internal clock by four. The divide-by-four stage, the 6-bit prescaler, and the 8-bit counter/timer form a synchronous 16-bit divide chain.

The counter/timer, prescaler, and associated mode registers are mapped into the register file as shown in Figure 3-22. This allows the software to treat the counter/timers as general-purpose registers, and eliminates the need for special instructions.

3.15 PRESCALERS AND COUNTER/TIMERS

The prescalers, PRE 0 (F5h) and PRE 1 (F3h), each consist of an 8-bit register and a 6-bit down-counter as shown in Figure 3-21. The prescaler registers are Write-Only registers. Reading the prescalers returns the value FFh. Figure 3-23 and Figure 3-24 show the prescaler registers.

The six most significant bits (D7,D6,D5,D4,D3,D2) of PRE0 or PRE1 hold the prescalers count modulo, a value from 1 to 64 decimal. The prescaler registers also contain control bits that specify T0 and T1 counting modes. These bits also indicate whether the clock source for T1 is internal or external. These control bits are discussed in detail throughout this chapter.

The counter/timer registers, T0 (F4h) and T1 (F2h), each consist of an 8-bit down-counter, a Write-Only register that holds the initial count value, and a Read-Only register that holds the current count value. The initial value can range from 1 to 256 decimal (01h, 02h,..., 00h).

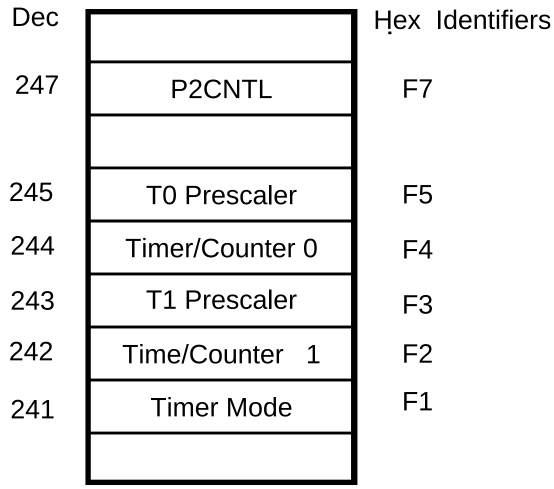


Figure 3-22. Counter/Timers Register Map

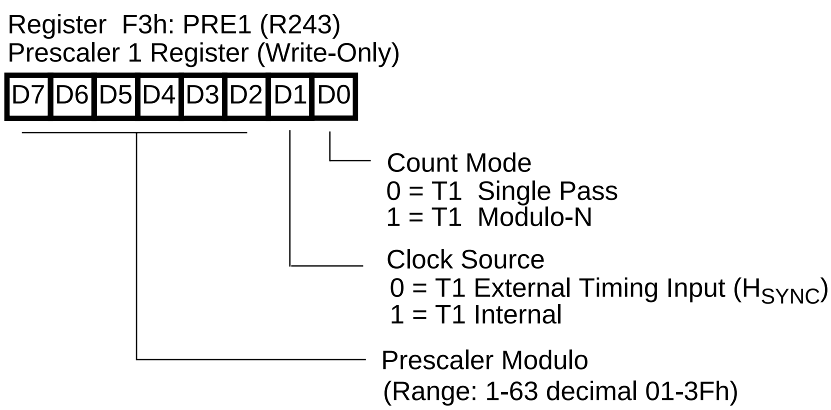


Figure 3-23. Prescaler 1 Register

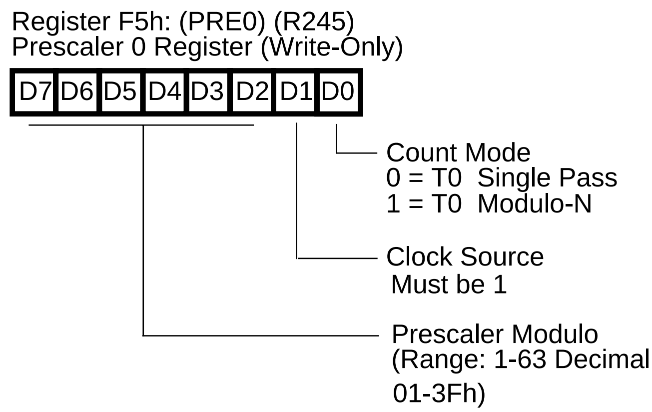


Figure 3-24. Prescaler 0 Register

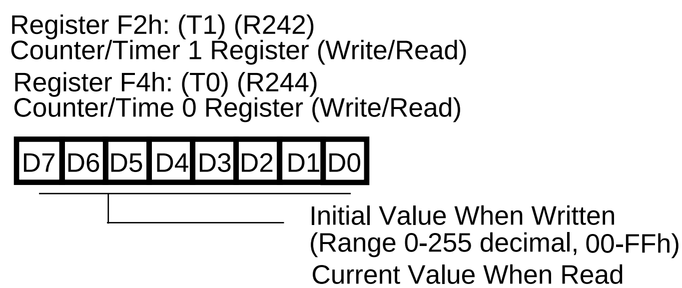


Figure 3-25. Counter/Timer 0 and 1 Registers

3.16 COUNTER/TIMERS OPERATION

Under software control, counter/timers are started and stopped via the Timer Mode Register (TMR, F1h) bits D3, D2, D1, D0. Each counter/timer is associated with a Load bit and an Enable Count bit.

3.16.1 Load and Enable Count Bits

Setting the Load bit (D0 for T0 and D2 for T1) transfers the initial value in the prescaler and the counter/timer registers into their respective down-counters. The next internal clock resets bits D0 and D2 to 0, readying the load bit for the next load operation. New values may be loaded into the down-counters at any time. If the counter/timer is running, it continues to do so and starts the count over with the new value. Therefore, the load bit actually functions as a software re-trigger.

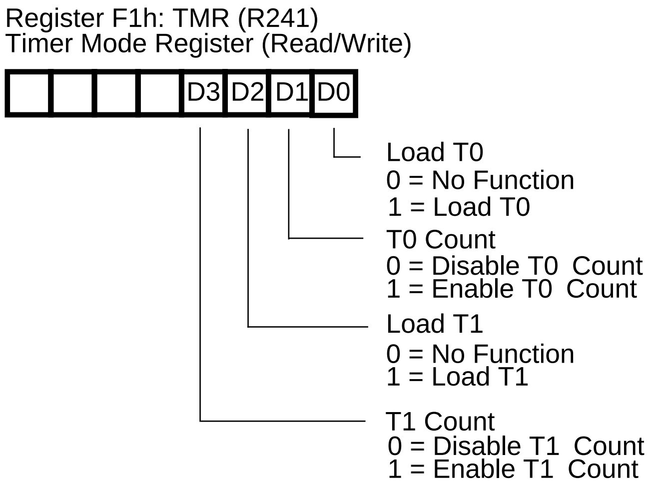


Figure 3-26. Timer Mode Register

The counter timers remain at rest as long as the Enable Count bits are 0. To enable counting, the Enable Count bit (D1 for T0 and D3 for T1) must be set to 1. Counting actually starts when the enable count bit is written by an instruction. The first decrement occurs four internal clock periods after the enable count bit has been set. If T1 is configured to use an external clock, the first decrement begins on the next clock period. The load and enable count bits can be set at the same time. For example, using the instruction:

OR TMR,#03h

sets both D0 and D1 of the TMR. This loads the initial values of PRE 0 and T0 into their respective counters and starts the count after the M2T2 (see Figure 3-28) machine state after the operand is fetched.

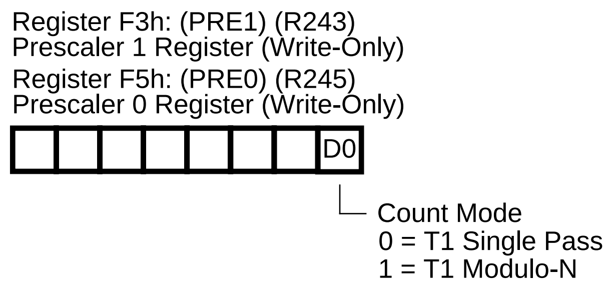


Figure 3-27. Starting The Count

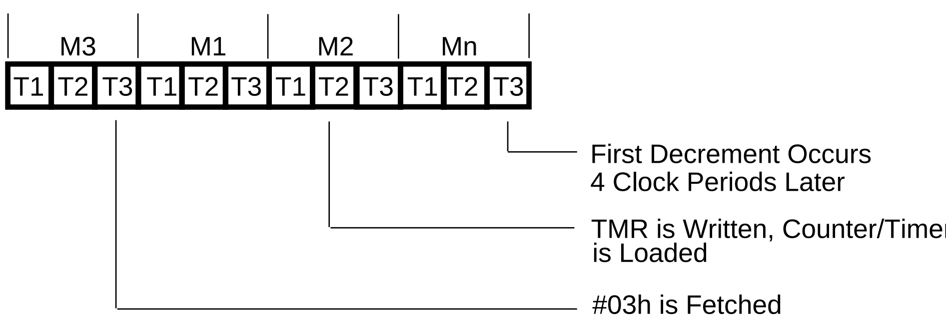


Figure 3-28. Counting Modes

3.16.2 Prescaler Operations

During counting, the programmed clock source drives the 6-bit Prescaler Counter. The counter is counted down from the value specified by bits of the corresponding Prescaler Register, PRE0 (7,6,5,4,3,2) or PRE1 (7,6,5,4,3,2). When the Prescaler Counter reaches its end of count, the initial value is reloaded and counting continues. The prescaler never actually reaches 0. For example, if the prescaler is set to divide-by-three, the count sequence is:

3–2–1–3–2–1–3–2–1–3.
..

Each time the prescaler reaches its end of count a carry is generated, which allows the counter/timer to decrement by one on the next timer clock input. When the counter/timer and the prescaler both reach the end-of-count, an interrupt request is generated (IRQ4 for T0, IRQ5 for T1). Depending on the counting mode selected, the counter/timer either rests with its value at 00h (Single-Pass Mode) or the initial value is automatically reloaded, and counting continues (Continuous Mode). The counting modes are controlled by PRE0 (0) and PRE1(0). A 0 written to this bit configures the counter for Single-Pass counting mode, while a 1 written to this bit configures the counter for Continuous Mode.

The counter/timer can be stopped at any time by setting the Enable Count bit to 0, and restarted by setting it back to 1. The counter/timer continues its count value at the time it was stopped. The current value in the counter/timer can be read at any time without affecting the counting operation.

Note: The prescaler registers are Write-Only and cannot be read.

New initial values can be written to the prescaler or the counter/timer registers at any time. These values are transferred to their respective down counters on the next load operation. If the counter/timer mode is Continuous, the next load occurs on the timer clock following an end-of-count. New initial values should be written before the desired load operation, since the prescalers always effectively operate in Continuous Mode.

The time interval (i) until end-of-count, is given by the equation:

$$i = t \text{ X } p \text{ X } v$$

in which:

t = four times the internal clock period.

The internal clock frequency defaults to the external clock source (XTAL, ceramic resonator, and others) divided by 2. Some microcontrollers allow this divisor to be changed via the Stop-Mode Recovery register. See the product data sheet for available clock divisor options.

Note that t is equal to eight divided-by-XTAL frequency of the external clock source for T (external clock mode only).

p = the prescaler value (1 – 63) for T0 and T1.

The minimum prescaler count of 1 is achieved by loading 000001XX. The maximum prescaler count of 63 is achieved by loading 111111XX.

v = the Counter/Timer value (1-255)

Minimum duration is achieved by loading 01h (1 prescaler output count), maximum duration is achieved by loading FFh (255 prescaler outputs counts).

The prescaler and counter/timer are true divide-by-n counters.

3.17 T_{IN} MODE

The Timer Mode Register TMR (F1h) is used to configure H_{SYNC} as T_{IN}. TMR (3), the enable count bit, must be set to 1 and initial values must be loaded into the down counters by setting the load bit, TMR (2), to a 1 before counting begins. In the descriptions of T_{IN} that follow, it is assumed the programmer has performed these operations. Initial values are automatically loaded in Trigger and Retrigger Modes so software loading is unnecessary.

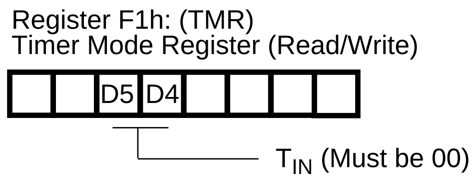


Figure 3-29. Timer Mode Register (T_{IN} Operation)

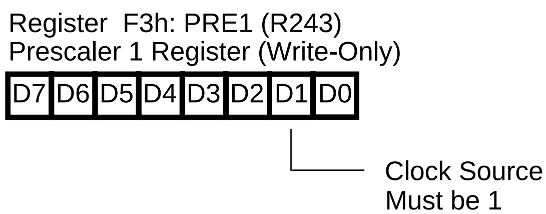


Figure 3-30. Prescaler 1 Register (T_{IN} Operation)

3.17.1 H_{SYNC} Clock Input Mode

The T_{IN} External Clock Input Mode (TMR bit 5 and bit 4 both set to 0) supports counting of external events, where an event is considered to be a High-to-Low transition on T_{IN}.

Note: See the product data sheet for the minimum allowed T_{IN} external clock input period (Tp T_{IN}).

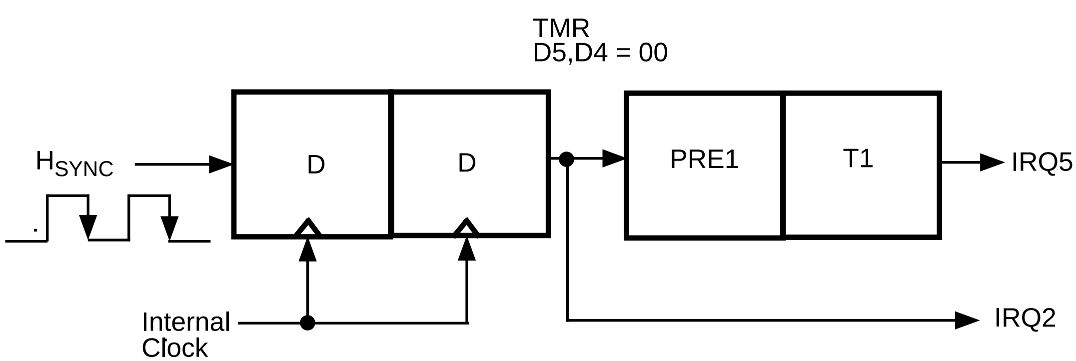


Figure 3-31. H_{SYNC} Clock Input Mode

3.18 COUNTER/TIMER RESET CONDITIONS

After a hardware reset, the counter/timers are disabled and the contents of the counter/timer and prescaler registers are undefined. However, the counting modes are configured for single-pass and the T clock source is set for external.

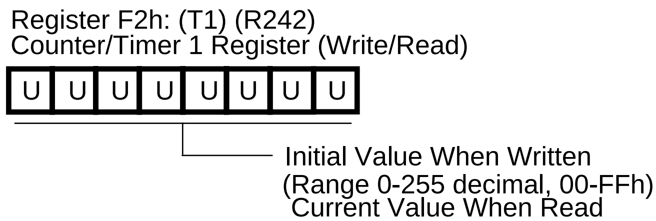


Figure 3-32. Counter/Timer 1 Register After Reset

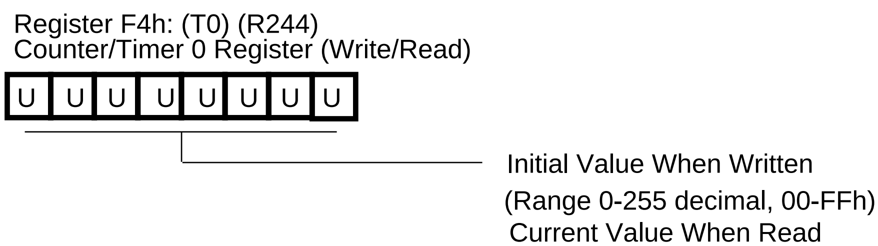


Figure 3-33. Counter/Timer 0 Register After Reset

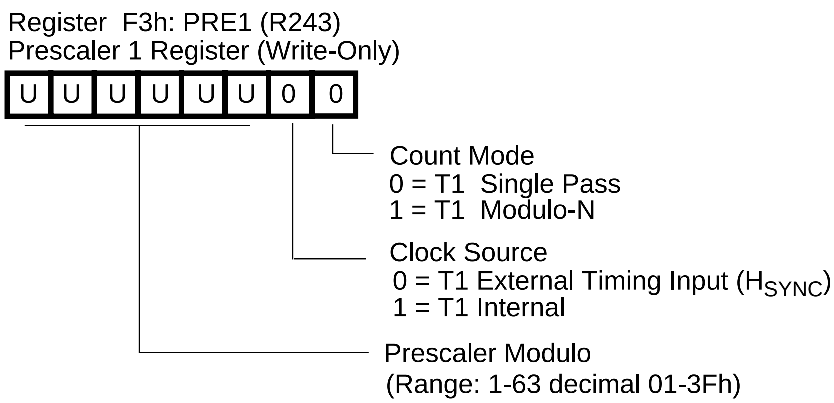


Figure 3-34. Prescaler 1 Register After Reset

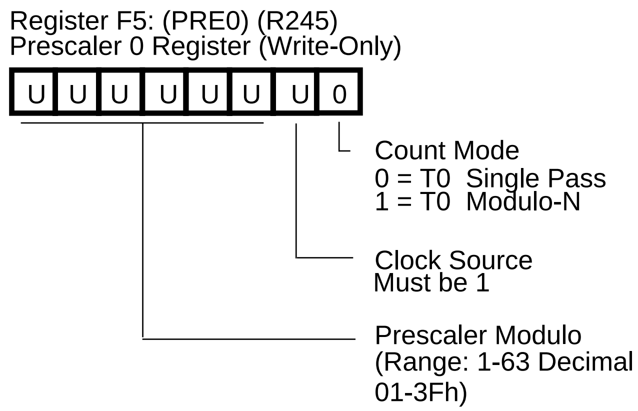


Figure 3-35. Prescaler 0 Register After Reset

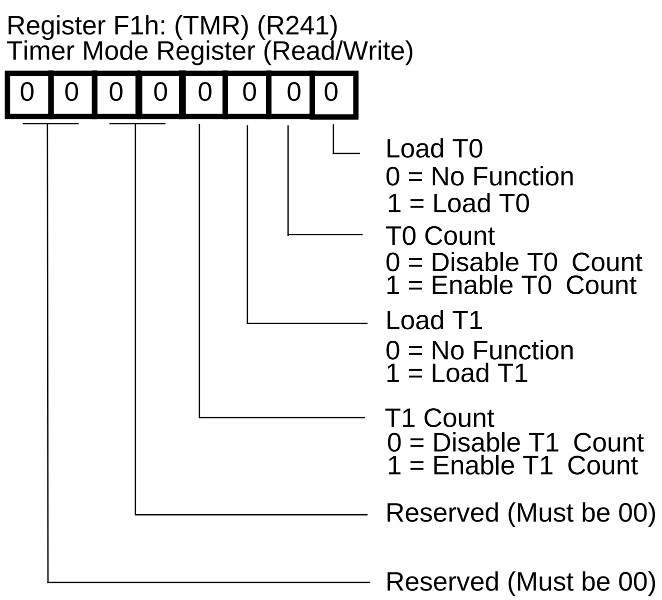


Figure 3-36. Timer Mode Register After Reset

3.19 INTERRUPTS

The microcontroller allows six different interrupts from a variety of sources, up to four external inputs, the on-chip counter/timer(s), software, and serial I/O peripherals. These interrupts can be masked and their priorities set by using the Interrupt Mask and the Interrupt Priority Registers. All six interrupts can be globally disabled by resetting the master interrupt enable, bit 7 in the interrupt mask register, with a Disable Interrupt (DI) instruction. Interrupts are globally enabled by setting bit 7 with an Enable Interrupt (EI) instruction.

Register	Hex	Identifier
Interrupt Mask	FBh	IMR
Interrupt Request	FAh	IRQ
Interrupt Priority	F9h	IPR

Figure 3-37. Interrupt Control Registers

There are three interrupt control registers: the Interrupt Request Register (IRQ), the Interrupt Mask register (IMR), and the Interrupt Priority Register (IPR). Figure 3-37 shows addresses and identifiers for the interrupt control registers. Figure 3-38 is a block diagram showing the Interrupt Mask and Interrupt Priority logic.

The Z8-MCU family supports both vectored and polled interrupt handling. Details on vectored and polled interrupts can be found later in this chapter.

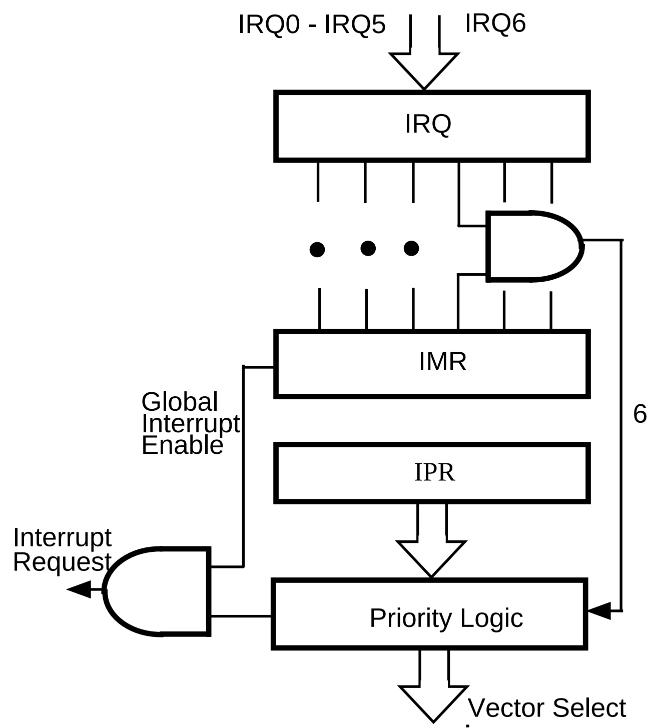


Figure 3-38. Interrupt Block Diagram

3.20 INTERRUPT SOURCES

Table 3-9 presents the interrupt types, sources, and vectors that are available.

Table 3-9. Interrupt Types, Sources, and Vectors

Name	Sources	Vector Location	Comments
IRQ0	IR Input	0,1	Edge Triggered; Internal
IRQ1	H_{SYNC} and V_{SYNC} Input	2,3	Edge Triggered; Internal. Generated at the start of every row and at the leading edge of the V_{SYNC} signal
IRQ2	P62	4,5	External (P62), Programmable Edge Triggered
IRQ3	P63	6,7	External (P63), Edge Triggered
IRQ 4	T0	8,9	Internal
IRQ5	T1	10,11	Internal

3.20.1 External Interrupt Source

External interrupt source involves IRQ3 and IRQ2, and can be generated by a transition on Port 63 and Port 62.

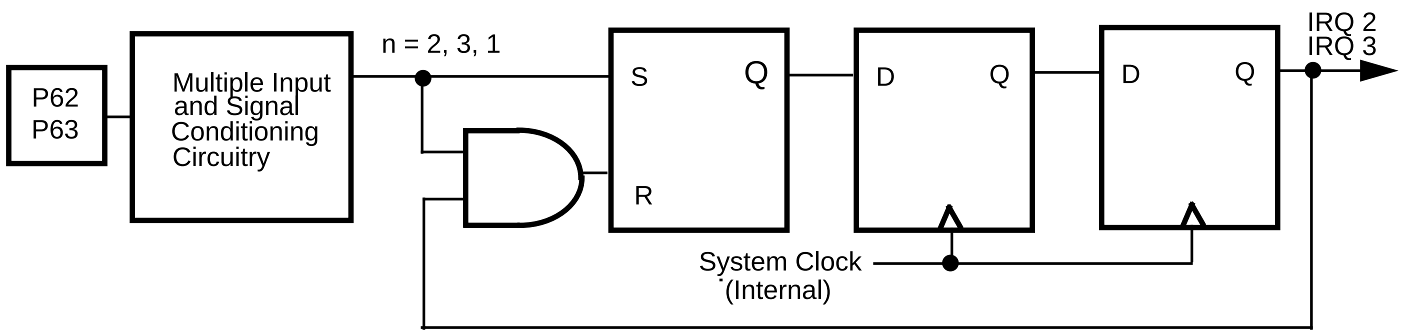


Figure 3-39. Interrupt Sources IRQ0-IRQ2 Block Diagram

When the port 6 pin (P63 and P62) transitions, the first flip-flop is set. The next two flip-flops synchronize the request to the internal clock and delay it by two internal clock periods. The output of the last flip-flop goes to D2 of the IRQ register for P62 and D3 for P63.

Note: Although interrupts are edge triggered, minimum interrupt request low and high times must be observed for proper operation. See AC Characteristics for exact timing requirements on external interrupt requests.

3.21 INTERRUPT REQUEST REGISTER LOGIC AND TIMING

Figure 3-40 shows the logic diagram for the Interrupt Request (IRQ) Register. The leading edge of the request sets the first flip-flop, which remains set until interrupt requests are sampled.

Requests are sampled internally during the last clock cycle before an op-code fetch (Figure 3-41). External requests are sampled two internal clocks earlier, due to the synchronizing flip-flops shown in Figure 3-40 and Figure 3-41.

At sample time the request is transferred to the second flip-flop in Figure 3-40, that drives the interrupt mask and priority logic. When an interrupt cycle occurs, this flip-flop will be reset only for the highest priority level that is enabled.

The user has direct access to the second flip-flop by reading and writing the IRQ Register. IRQ is read by specifying it as the source register of an instruction and written by specifying it as the destination register.

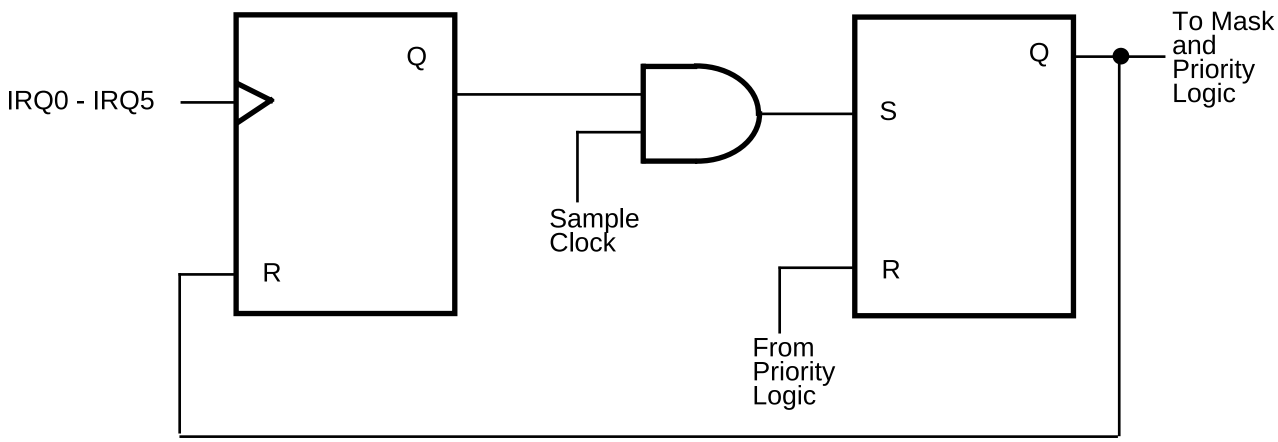


Figure 3-40. IRQ Register Logic

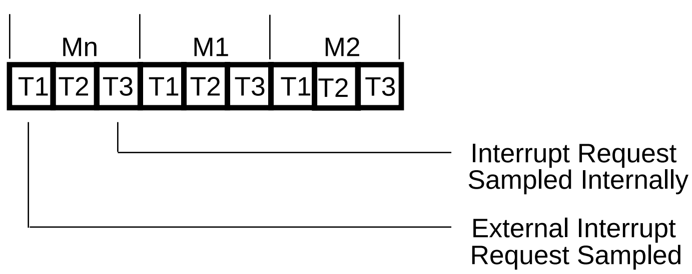


Figure 3-41. Interrupt Request Timing

3.22 INTERRUPT INITIALIZATION

After reset, all interrupts are disabled and must be initialized before vectored or polled interrupt processing can begin. The Interrupt Priority Register (IPR), Interrupt Mask Register (IMR), and Interrupt Request Register (IRQ) must be initialized, in that order, to start the interrupt process. However, IPR need not be initialized for polled processing.

3.22.1 Interrupt Priority Register Initialization

An Interrupt Priority Register (IPR) initialization is a Write-Only register that sets priorities for the vectored interrupts in order to resolve simultaneous interrupt requests. (There are 48 sequence possibilities for interrupts.)

The six interrupt levels IRQ0-IRQ5 are divided into three groups of two interrupt requests each. One group contains IRQ3 and IRQ5. The second group contains IRQ0 and IRQ2, while the third group contains IRQ1 and IRQ4.

Priorities can be set both within and between groups as shown in Table 3-10 and Table 3-11. Bits 1, 2, and 5 define the priority of the individual members within the three groups. Bits 0, 3, and 4 are encoded to define six priority orders between the three groups. Bits 6 and 7 are

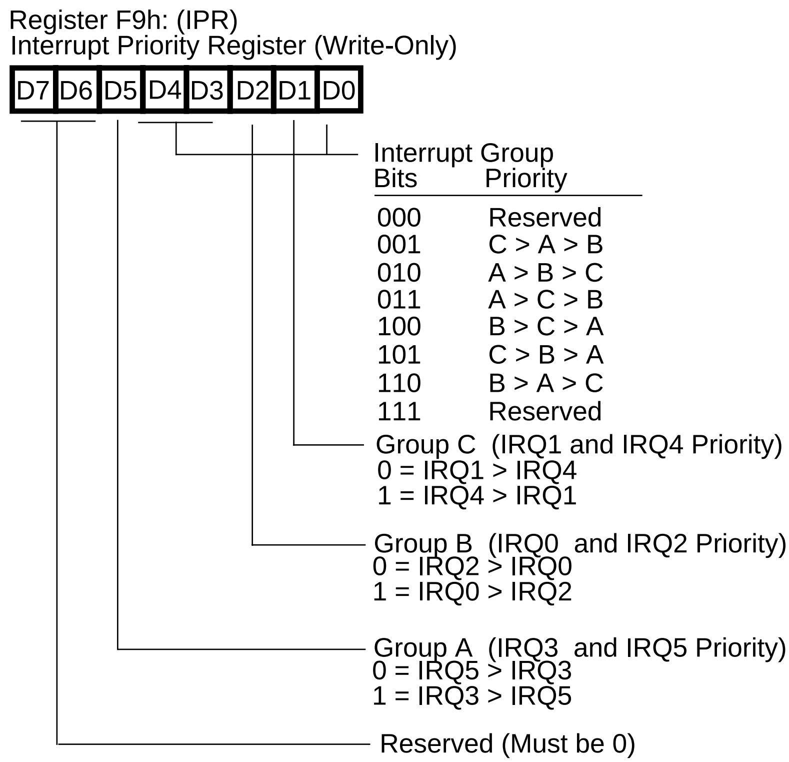


Figure 3-42. Interrupt Priority Register

Table 3-10. Interrupt Priority

Group	Bit	Value	Priority Highest	Lowest
C	1	0	IRQ1	IRQ4
		1	IRQ4	IRQ1
B	2	0	IRQ2	IRQ0
		1	IRQ0	IRQ2
A	5	0	IRQ5	IRQ3
		1	IRQ3	IRQ5

Table 3-11. Interrupt Group Priority

Bit Pattern			Group Priority		
Bit 4	Bit 3	Bit 0	High	Medium	Low
0	0	0	Not Used		
0	0	1	C	A	B
0	1	0	A	B	C
0	1	1	A	C	B
1	0	0	B	C	A
1	0	1	C	B	A
1	1	0	B	A	C
1	1	1	Not Used		

3.22.2 Interrupt Mask Register Initialization

An Interrupt Mask Register (IMR) initialization individually or globally enables or disables the six interrupt requests. When bits 5,4,3,2,1,0 are set to 1, the corresponding interrupt requests are enabled. Bit 7 is the master enable and must be set before any of the individual interrupt requests can be recognized. Resetting bit 7 globally disables all the interrupt requests. Bit 7 is set and reset by the EI and DI instructions. It is automatically reset during an interrupt service routine and set following the execution of an Interrupt Return (IRET) instruction.

Note: Bit 7 must be reset by the DI instruction before the contents of the Interrupt Mask Register or the Interrupt Priority Register are changed except:

- Immediately after a hardware reset.
- Immediately after executing an interrupt service routine and before IMR (7) has been set by any instruction.

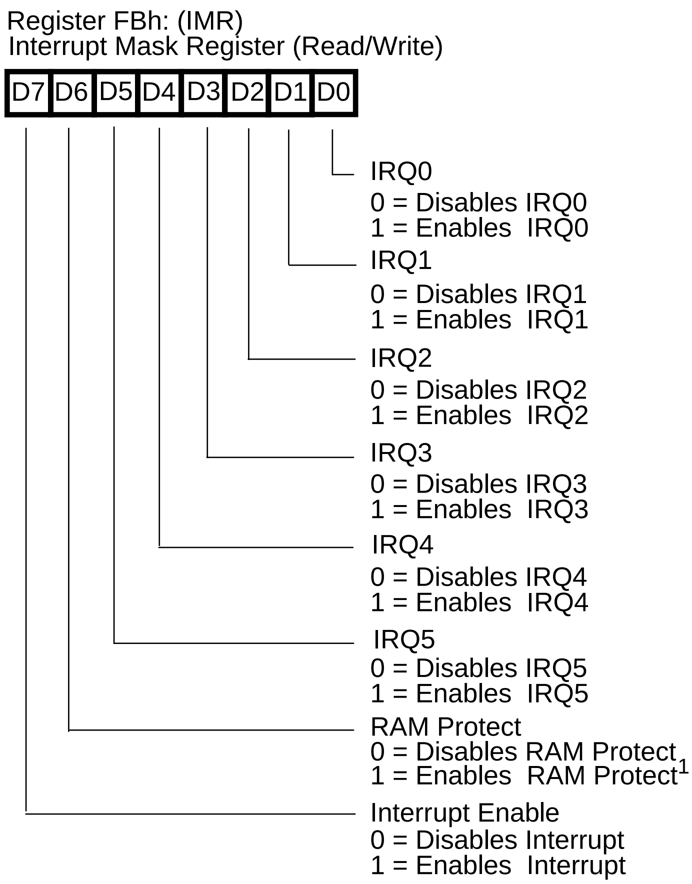


Figure 3-43. Interrupt Mask Register

Notes:
1. The RAM Protect option is selected at ROM mask submission time or at EPROM program time. If not selected or not an available option, this bit is reserved and must be 0.

3.22.3 Interrupt Request Register Initialization

An Interrupt Request Register (IRQ) (Figure 3-44) is a read/write register that stores the interrupt requests for both vectored and polled interrupts. When an interrupt is made on any of the six, the corresponding bit position in the register is set to 1. Bit 0 to bit 5 are assigned to interrupt requests IRQ0 to IRQ5, respectively.

Whenever Power-On Reset (POR) is executed, the IRQ register is reset to 00h and disabled. Before the IRQ register will accept requests, it must be enabled by executing an ENABLE INTERRUPTS (EI) instruction.

Note: Setting the Global Interrupt Enable bit in the Interrupt Mask Register (IMR, bit 7) does not enable the IRQ. Execution of the EI instruction is required (Figure 3-44).

For polled processing, IRQ must still be initialized by an EI instruction.

To properly initialize the IRQ register, the following code is provided:

```
CLR  IMR    //make sure disabled vectored
            interrupts.
EI          //enable IRQ register otherwise
            read only.
            //not needed if interrupts were
            previously enabled.
DI          //disable interrupt heading.
```

Note: An IRQ is always cleared to 00h and is read only until the first EI instruction which enables the IRQ to be read/write.

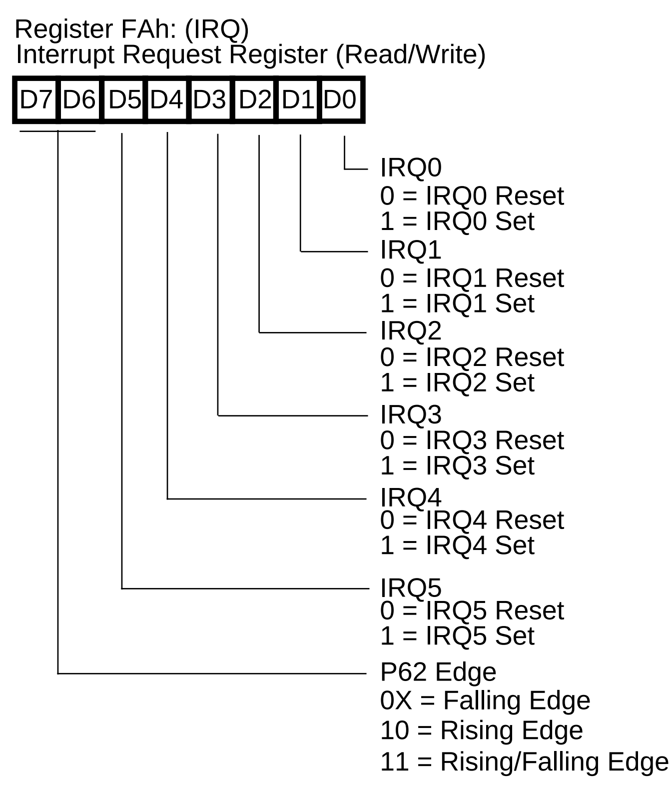


Figure 3-44. Interrupt Request Register

The functions of the IRQs are as follows:

Table 3-12. IRQ Function Summary

IRQ	Function
IRQ0	IR input
IRQ1	HV_{SYNC} input
IRQ2	P62 input[1]
IRQ3	P63 input[1]
IRQ4	T0 internal timer
IRQ5	T1 internal timer

Note:
1. P62 and P63 must be configured as input if used as an interrupt source.

Data bits 6 and 7 set the P62 edge. Some coding is required to clear P62 for input, for example:

- To select Rising Edge for P62 interrupt:

```
.....
.....
DI                ; disable all interrupts
OR IRQ    #%80    ; enable rising edge for
                  ; P62 interrupt
AND IRQ   #%FB    ; clear IRQ2 (P62
                  ; interrupt),
                  ;  keep other IRQ's bits
                  ; untouched
 EI               ; enable interrupts
.....
.....
```

- To select Rising and Falling Edge for P62 interrupt:

```
.....
.....
DI                ; disable all interrupts
OR IRQ    #%C0    ; enable rising and falling
                  ; edge for
                  ; P62 interrupt
AND IRQ   #%FB    ; clear IRQ2 bit (P62
                  ; interrupt),keep other
                  ; IRQ's bits untouched
EI                ; enable interrupts
.....
.....
```

The IMR is cleared before the IRQ enabling sequence to insure no unexpected interrupts occur when EI is executed. This code sequence should be executed prior to programming the application required values for IPR and IMR.

Note: IRQ bits 6 and 7 are device dependent. When reserved, the bits are not used and will return a 0 when read. When used as the Interrupt Edge select bits, the configuration options are as shown in the following table.

Table 3-13. IRQ Register Configuration

IRQ D7	D6	Interrupt Edge P62
0	0	Falling
0	1	Falling
1	0	Rising
1	1	Rising/Falling

The proper sequence for programming the interrupt edge select bits is (assumes IPR and IMR have been previously initialized):

```
DI                          ;Inhibit all
                            ; interrupts
                            ;until input
                            ; edges are
                            ;configured
OR    IRQ,#XX 000000B       ;Configure
                            ; interrupt
                            ;do not disturb
                            ;edges as
                            ; needed -
                            ;IRQ 0-5.
EI                          ;Re-enable
                            ; interrupts.
```

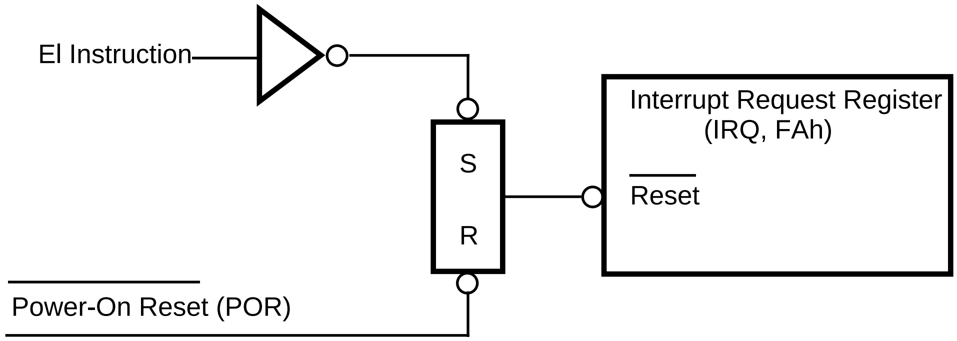


Figure 3-45. IRQ Reset Functional Logic Diagram

3.23 IRQ SOFTWARE INTERRUPT GENERATION

An IRQ can be used to generate software interrupts by specifying an IRQ as the destination of any instruction referencing the Standard Register File. These Software Interrupts (SWI) are controlled in the same manner as hardware generated requests (in other words, the IPR and the IMR control the priority and enabling of each SWI level).

To generate a SWI, the desired request bit in the IRQ is set as follows:

OR IRQ, #NUMBER

where the immediate data, NUMBER, has a 1 in the bit position corresponding to the level of the SWI desired. For example, if an SWI is desired on IRQ5, NUMBER would have a 1 in bit 5:

OR IRQ, #00100000B

With this instruction, if the interrupt system is globally enabled, IRQ5 is enabled, and there are no higher priority pending requests, control is transferred to the service routine pointed to by the IRQ vector.

3.24 VECTORED PROCESSING

Each interrupt level has its own vector. When an interrupt occurs, control passes to the service routine pointed to by the interrupt's vector location in program memory. The sequence of events for vectored interrupts is as follows:

- PUSH the Program Counter (PC) lower byte on to the stack
- PUSH the PC upper byte on to the stack
- PUSH FLAGS on to the stack
- Fetch the upper byte of the vector
- Fetch the lower byte of the vector
- Branch to the service routine specified by the vector

Figure 3-46 and Figure 3-47 illustrate the vectored interrupt operation.

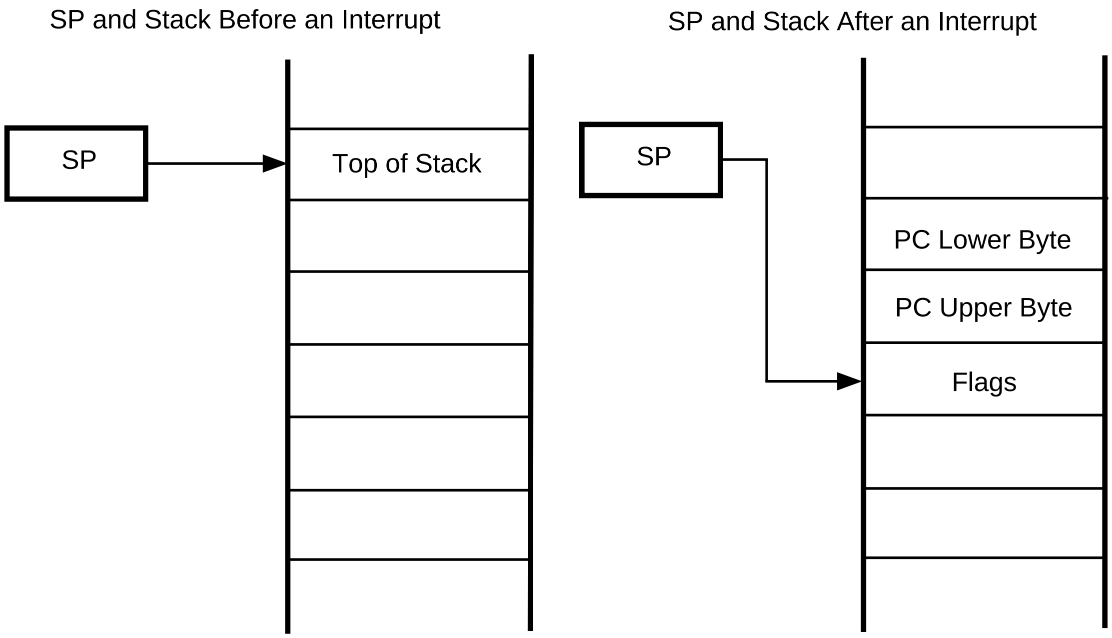


Figure 3-46. Effects of an Interrupt on the Stack

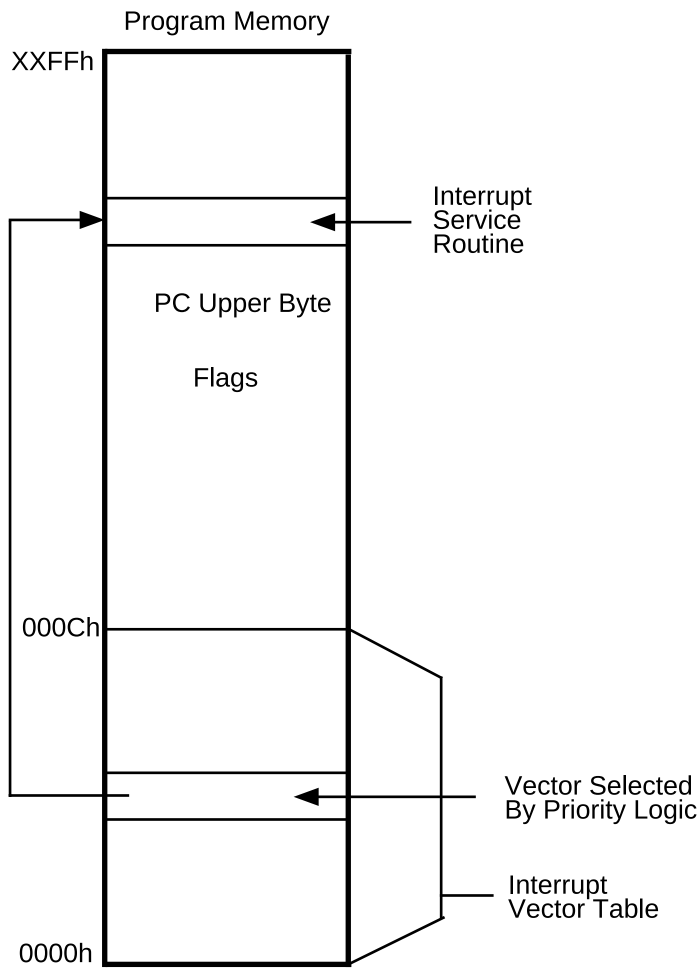


Figure 3-47. Interrupt Vectors in Memory

3.24.1 Vectored Interrupt Cycle Timing

The interrupt acknowledge cycle time is 24 internal clock cycles. In addition, two internal clock cycles are required for the synchronizing flip-flops. The maximum interrupt recognition time is equal to the number of clock cycles required for the longest executing instruction present in the user program (assumes worst case condition of interrupt sampling, Figure 3-48, just prior to the interrupt occurrence). To calculate the worst case interrupt latency (maximum time required from interrupt generation to fetch of the first instruction of the interrupt service routine), sum these components:

Worst Case Interrupt Latency ≈ 24 INT CLK (interrupt acknowledge time) + # TpC of longest instruction present in the user's application program + 2 TpC (internal synchronization time).

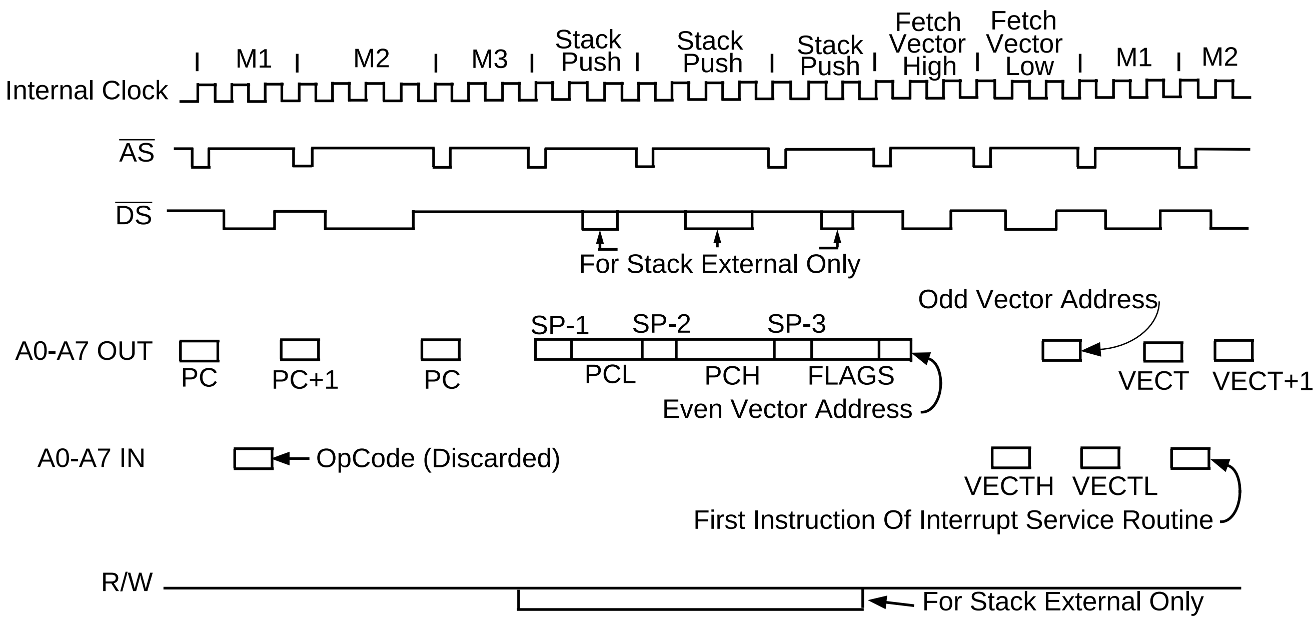


Figure 3-48. Interrupt Acknowledge Timing

3.24.2 Nesting of Vectored Interrupts

Nesting of vectored interrupts allows higher priority requests to interrupt a lower priority request. To initiate vectored interrupt nesting, do the following during the interrupt service routine:

- Push the old IMR on to the stack.
- Load IMR with a new mask to disable lower priority interrupts.
- Execute EI instruction.
- Proceed with interrupt processing.
- After processing is complete, execute DI instruction.
- Restore the IMR to its original value by returning the previous mask from the stack.
- Execute IRET.

Depending on the application, some simplification of the above procedure may be possible.

3.25 POLLED PROCESSING

Polled interrupt processing is supported by masking off the IRQ to be polled. This is accomplished by clearing the corresponding bits in the IMR.

To enable any interrupt, first the interrupt mechanism must be engaged with an EI instruction. If only polled interrupts are to be serviced, execute:

```
EI ;Enable interrupt mechanism
DI ;Disable vectored interrupts
```

To initiate polled processing, check the bits of interest in the IRQ using the Test Under Mask (TM) instruction. If the bit is set, call or branch to the service routine. The service routine services the request, resets its Request Bit in the IRQ, and branches or returns back to the main program. An example of a polling routine is as follows:

```
    TM   IRQ, #MASKA      ;Test for request
    JR   Z, NEXT          ;If no request go to
                          ;NEXT
    CALL SERVICE          ;If request is there,
                          ;then service it
NEXT:
    .
    .
    .
SERVICE:                  ;Process Request
    .
    .
    .
    AND IRQ, #MASKB       ;Clear Request Bit
    RET                   ;Return to NEXT
```

In this example, if IRQ2 is being polled, MASKA

3.26 INTERRUPT RESET CONDITIONS

At Reset, all bits in IPR are undefined.

In IMR, bit 7 is 0 and bits 0-6 are undefined. The IRQ register is reset and held in that state until an enable interrupt (EI) instruction is executed.

3.27 POWER-DOWN HALT-MODE OPERATION

The Halt Mode suspends instruction execution and turns off the internal CPU clock. The on-chip oscillator circuit remains active so the internal clock continues to run and is applied to the counter/timer(s) and interrupt logic.

To enter the Halt Mode, it is necessary to first flush the instruction pipeline to avoid suspending execution in mid-instruction. To do this, the application program must execute a NOP instruction (opcode = FFh) immediately before the Halt instruction (opcode 7Fh), that is,

```
FF   NOP    ;clear the instruction pipeline
7F   Halt   ;enter Halt Mode
```

The Halt Mode is exited by interrupts, either externally or internally generated. Upon completion of the interrupt service routine, the user program continues from the instruction after Halt.

The Halt Mode may also be exited via a POR/Reset activation or a Watch-Dog Timer (WDT) timeout. (See the product data sheet for WDT availability.) In this case, program execution restarts at the reset-restart address 000Ch.

To further reduce power consumption in the Halt Mode, some Z8-family devices allow dynamic internal clock scaling. Clock scaling may be accomplished on the fly by reprogramming bit 0 and/or bit1 of the Stop-Mode Recovery register (SMR).

Note: Internal clock scaling directly effects Counter/Timer operation — adjustment of the prescaler and downcounter values may be required. To determine the actual Halt mode current (ICC1) value for the various optional modes available, see the selected microcontroller device’s product specification.

3.28 STOP-MODE OPERATION

The Stop Mode provides the lowest possible device standby current. This instruction turns off the on-chip oscillator and internal system clock.

To enter the Stop Mode, it is necessary to first flush the instruction pipeline to avoid suspending execution in mid-instruction. To do this, the application program must execute a NOP instruction (opcode=FFh) immediately before the Stop instruction (opcode=6Fh), that is,

```
FF   NOP    ;clear the instruction pipeline
6F   Stop   ;enter Stop Mode
```

The Stop Mode is exited by any one of the following resets: Power-On Reset activation, WDT timeout (if available), or a Stop-Mode Recovery source. Upon reset generation, the processor always restarts the application program at address 000Ch.

POR/Reset activation is present on all Z8-base devices and is implemented as a reset pin and/or an on-chip power on reset circuit.

Some microcontrollers allow for the on-chip WDT to run in the Stop Mode. If so activated, the WDT timeout generates a Reset some fixed time period after entering the Stop Mode.

Note: Stop-Mode Recovery (SMR) by the WDT increases the Stop Mode standby current (ICC2). This is due to the WDT clock and divider circuitry that is now enabled and running to support this recovery mode. See the product data sheet for actual ICC2 values.

All Z8-microcontroller bases provide some form of dedicated Stop-Mode Recovery (SMR) circuitry. Two SMR methods are implemented — a single-fixed input pin or a flexible, programmable set of inputs. The selected Z8-base product specification should be reviewed to determine the SMR options available for use.

Note: For devices that support SPI, the Slave mode compare feature also serves as a SMR source.

In the simple case, a Low level applied to input pin P27 triggers a SMR. To use this mode, pin P27 (I/O Port 2, bit 7) must be configured as an input before the Stop Mode is entered. The Low level on P27 must meet a minimum pulse width TWSM. (See the product data sheet to trigger the device Reset Mode.) Some microcontrollers provide multiple SMR input sources. The desired SMR source is selected via the SMR Register.

Note: Use of specialized SMR modes (P27 input or SMR register based) or the WDT timeout (only when in the Stop Mode) provide a unique reset operation. Some control registers are initialized differently for a SMR/WDT triggered POR than a standard reset operation. See the product specification (register file map) for exact details.

To determine the actual Stop Mode current (ICC2) value for the optional SMR modes available, see the selected Z8 device’s product data sheet.

Note: The Stop Mode current (ICC2) is minimized when:

- V_{CC} is at the low end of the device’s operating range.
- WDT is Off in the Stop Mode.
- Output current sourcing is minimized.
- All inputs (digital and analog) are at the low or high rail voltages.

3.29 STOP-MODE RECOVERY REGISTER

This register selects the clock divide value and determines the mode of Stop-Mode Recovery. All bits are Write-Only, except bit 7, that is Read-Only. Bit 7 is a flag bit that is hardware set on the condition of Stop-Mode Recovery, and reset by a power-on cycle. Bit 6 controls whether a Low level or a High level is required from the recovery source. Bit 5 controls the reset delay after recovery. Bits 2, 3, and 4, of the SMR register, specify the source of the Stop-Mode Recovery signal. Bits 0 and 1 control internal clock divider circuitry. The SMR is located in bank F of the expanded register file at address 0Bh.

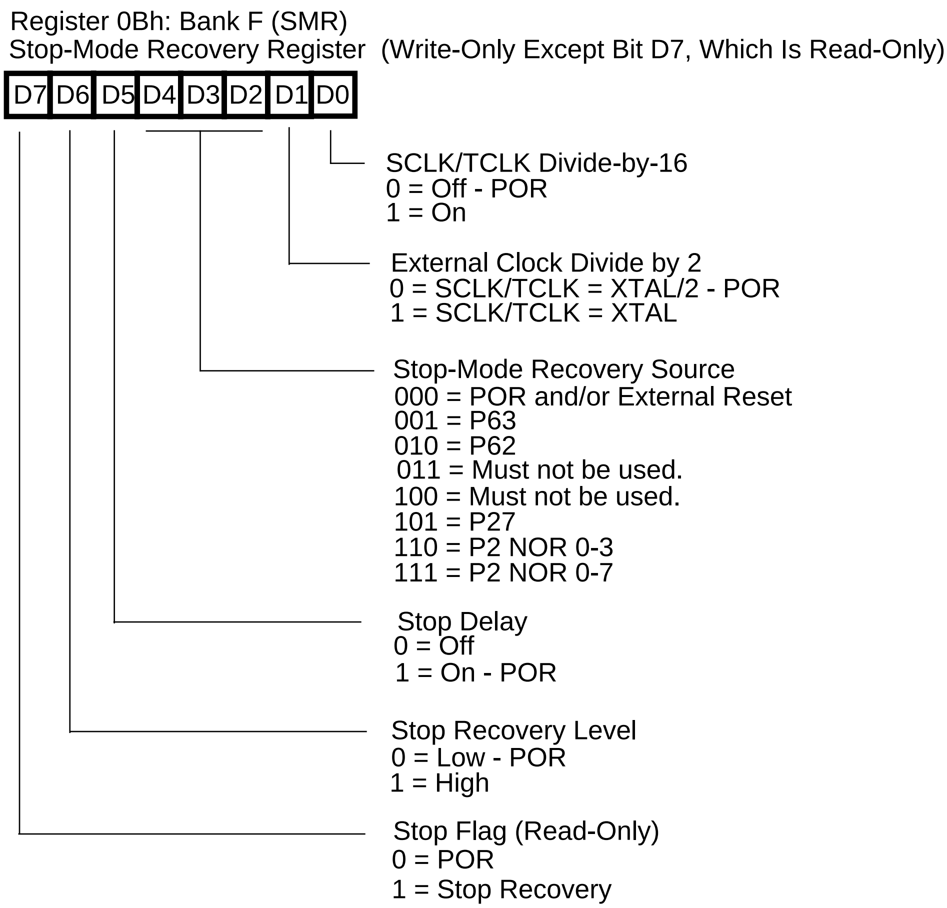


Figure 3-49. Stop-Mode Recovery Register

- **SCLK/TCLK Divide-by-16 Select (DO):** This bit of the SMR controls a divide-by-16 prescaler of SCLK/TCLK. The purpose of this control is to selectively reduce device power consumption during normal processor execution (SCLK control) and/or Halt Mode (where TCLK sources counter/timers and interrupt logic).

- **External Clock Divide-by-Two (D1):** This bit can eliminate the oscillator divide-by-two circuitry. When this bit is 0, the System Clock (SCLK) and Timer Clock (TCLK) are equal to the external clock frequency divided by two. The SCLK/TCLK is equal to the external clock frequency when this bit is set (D1=1). Using this bit together with D7 of PCON helps further lower EMI (D7 (PCON) =0, D1 (SMR) =1). The default setting is zero.

- **Stop-Mode Recovery Source (D2, D3, and D4):** These three bits of the SMR specify the wake-up source of the Stop-Mode recovery (Table 3-14 and Figure 3-50).

Table 3-14. Stop-Mode Recovery Source

SMR: 432 D4	D3	D2	Operation Description of Action
0	0	0	POR and/or external reset recovery
0	0	1	P63 transition
0	1	0	P62 transition (not in Analog Mode)
1	0	1	P27 transition
1	1	0	Logical NOR of P20 through P23
1	1	1	Logical NOR of P20 through P27

- **Stop-Mode Recovery Delay Select (D5):** This bit, if High, enables the T_{POR} $\overline{\text{Reset}}$ delay after Stop-Mode Recovery. The default configuration of this bit is 1. If the fast wake up is selected, the Stop-Mode Recovery source is kept active for at least 5 TpC.

- **Stop-Mode Recovery Level Select (D6):** A 1 in this bit position indicates that a High level on any one of the recovery sources wakes the microcontroller from Stop Mode. A 0 indicates Low-level recovery. The default is 0 on POR (Figure 3-50).

- **Cold or Warm Start (D7):** This bit is set by the device upon entering Stop Mode. A 0 in this bit (cold) indicates that the device reset by POR/WDT Reset. A 1 in this bit (warm) indicates that the device awakens by a SMR source.

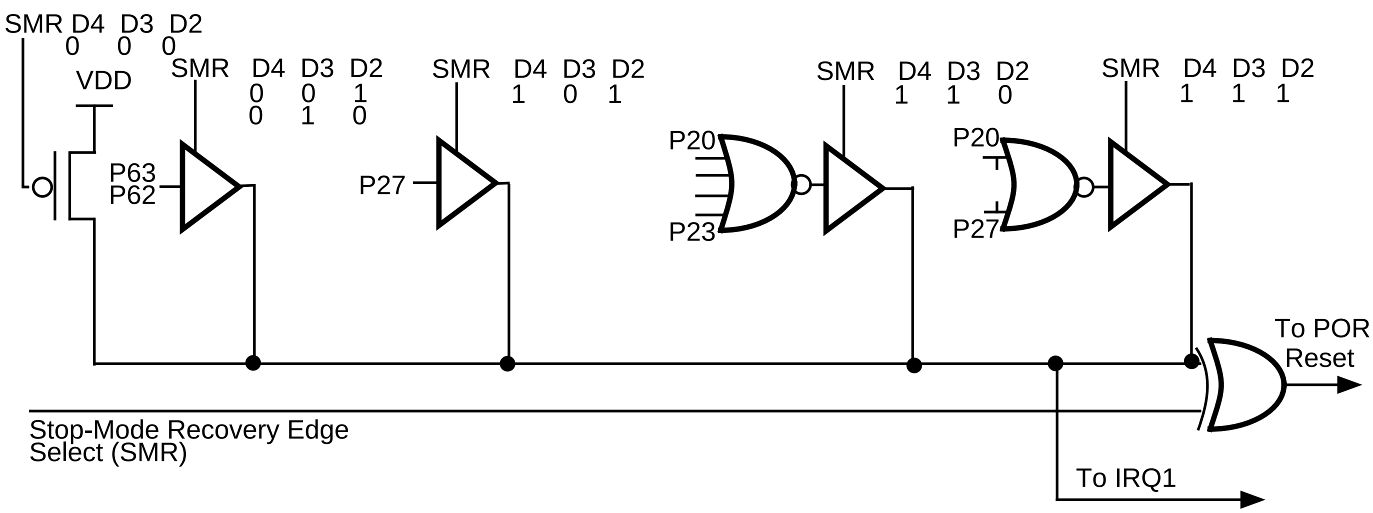


Figure 3-50. Stop-Mode Recovery Source/Level Select

Note: If P62 is used as a SMR source, the digital mode of operation must be selected prior to entering the Stop Mode.

3.30 ADDRESSING MODES

Six addressing modes are available:

- Register (R)
- Indirect Register (IR)
- Indexed (X)
- Direct (D)
- Relative (RA)
- Immediate (IM)

With the exception of immediate data and condition codes, all operands are expressed as register file, program memory, or data memory addresses. Registers are accessed using 8-bit addresses in the range of 00h-FFh. The program memory or data memory is accessed using 16-bit addresses (register pairs) in the range of 0000h-FFFFh.

Working registers are accessed using 4-bit addresses in the range of 0-15 (0h-Fh). The address of the register being accessed is formed by the combination of the upper four bits in the register pointer (R253) and the 4-bit working register address supplied by the instruction.

Registers can be used in pairs to designate 16-bit values or memory addresses. A Register Pair must be specified as an even-numbered address in the range of 0, 2,...., 14 for working registers, or 4, 6,....238 for actual registers.

In the following definitions of addressing modes, the use of ‘register’ can also imply register pair, working register, or working register pair, depending on the context.

Note: See the product data sheet for exact program, data, and register memory types and address ranges available.

3.31 REGISTER ADDRESSING

In 8-bit Register Addressing (R) mode, the operand value is equivalent to the contents of the specified register or register pair.

In the register addressing (Figure 3-51), the destination and/or source address specified corresponds to the actual register in the register file.

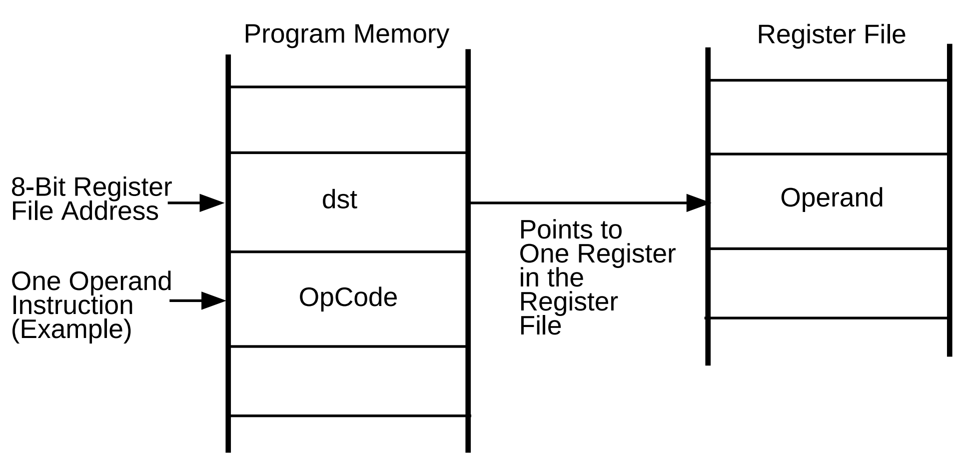


Figure 3-51. 8-Bit Register Addressing

In 4-bit Register Addressing (Figure 3-52), the destination and/or source addresses point to the working register within the current working register group. This 4-bit address is combined with the upper 4 bits of the register pointer to form the actual 8-bit address of the affected register.

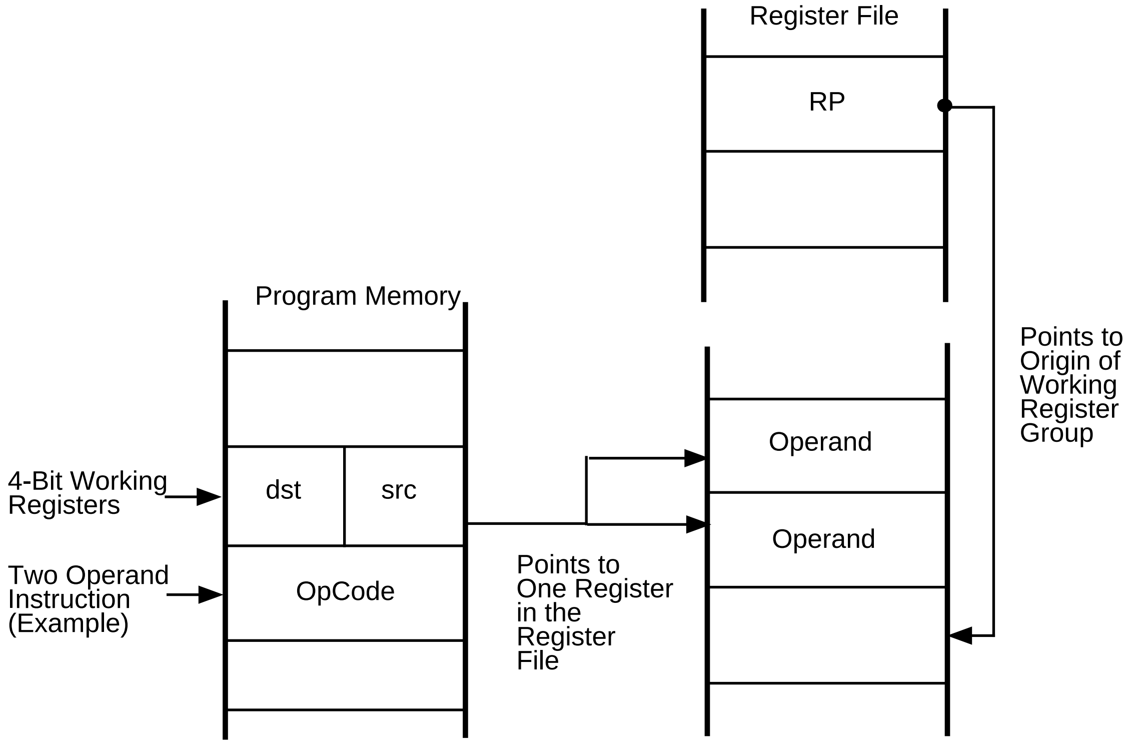


Figure 3-52. 4-Bit Register Addressing

3.32 INDIRECT REGISTER ADDRESSING

In the Indirect Register Addressing Mode, the contents of the specified register are equivalent to the address of the operand (Figure 3-53 and Figure 3-54).

Depending upon the instruction selected, the specified register contents points to a register, program memory, or an external data memory location.

When accessing program memory or external data memory, register pairs or working register pairs are used to hold the 16-bit addresses.

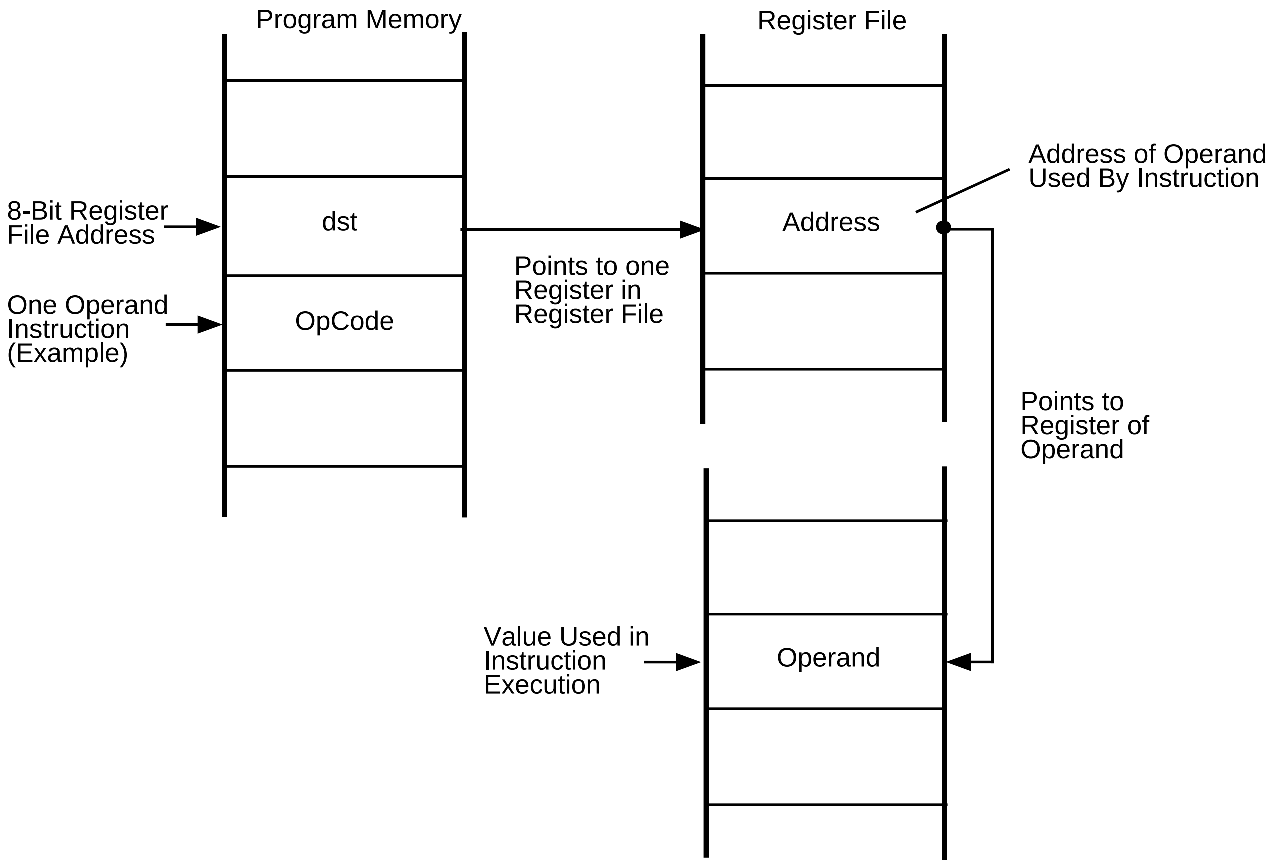


Figure 3-53. 8-Bit Indirect Register Addressing

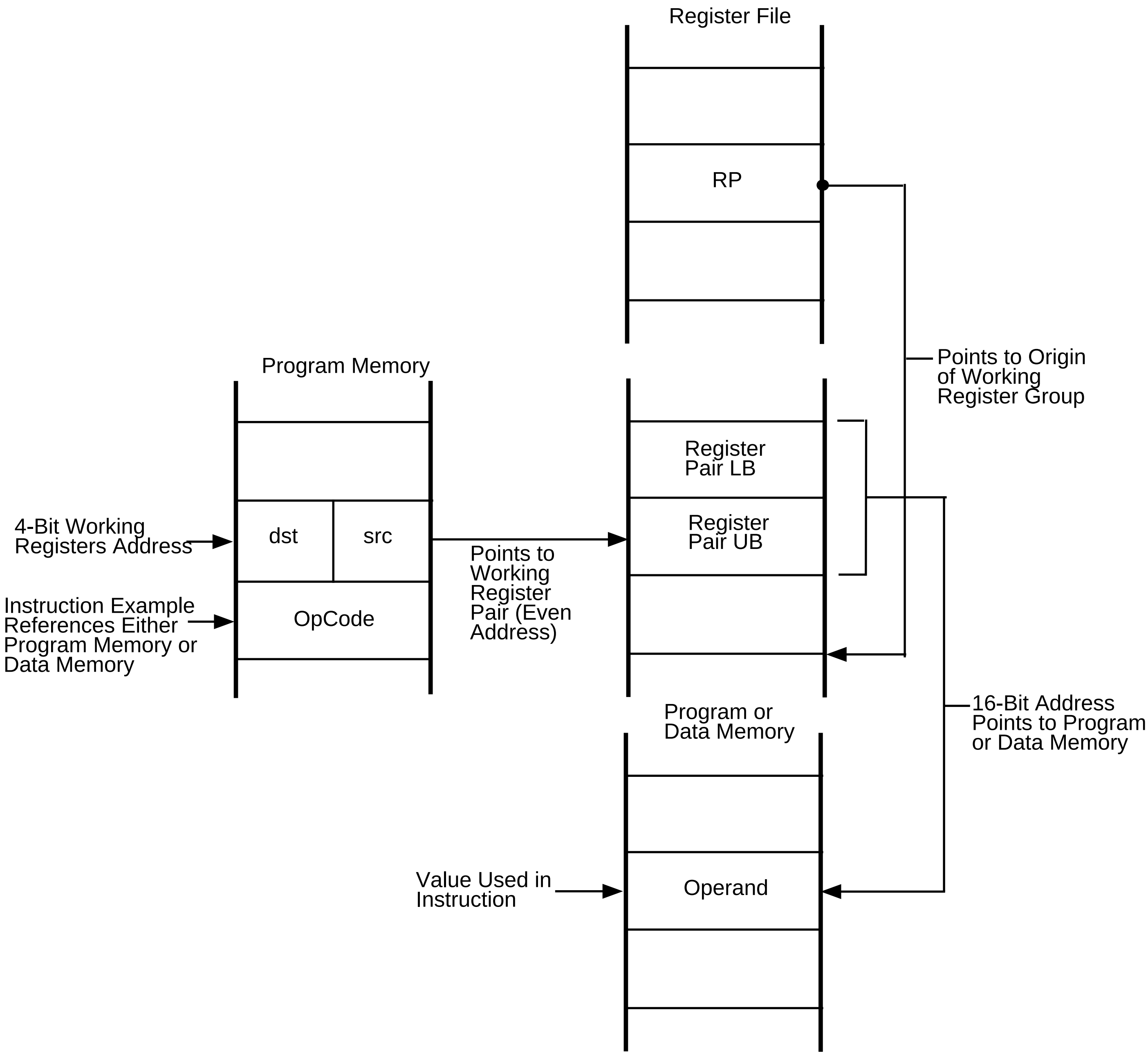


Figure 3-54. 4-Bit Indirect Register Addressing

3.33 INDEXED ADDRESSING

The Indexed Addressing Mode is used only by the Load (LD) instruction. An indexed address consists of a register address offset by the contents of a designated working register (the Index). This offset is added to the register address to obtain the address of the operand. Figure 3-55 illustrates this addressing convention.

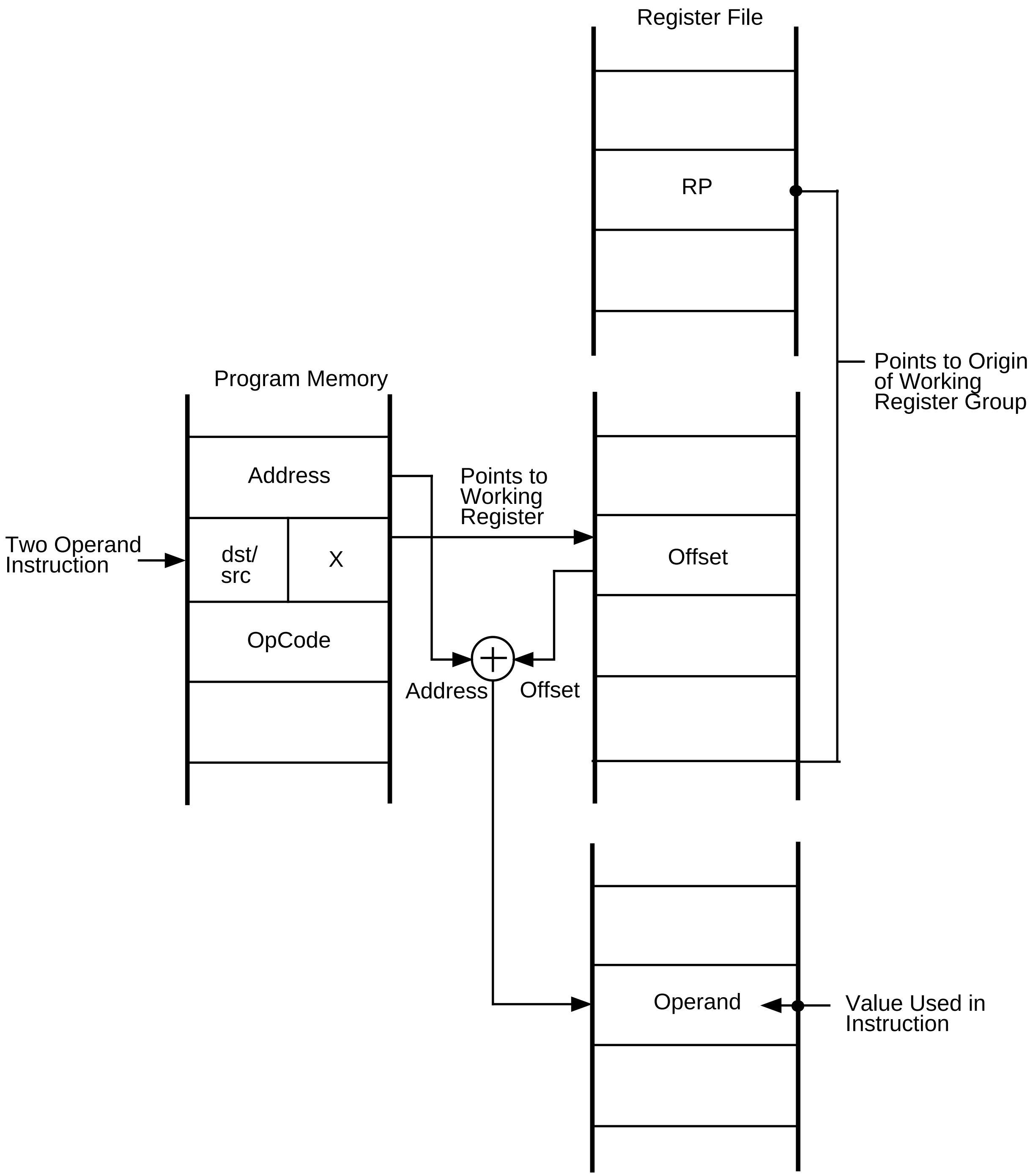


Figure 3-55. Indexed Register Addressing

3.34 DIRECT ADDRESSING

The Direct Addressing mode, as shown in Figure 3-56, specifies the address of the next instruction to be executed. Only the Conditional Jump (JP) and Call (CALL) instructions use this addressing mode.

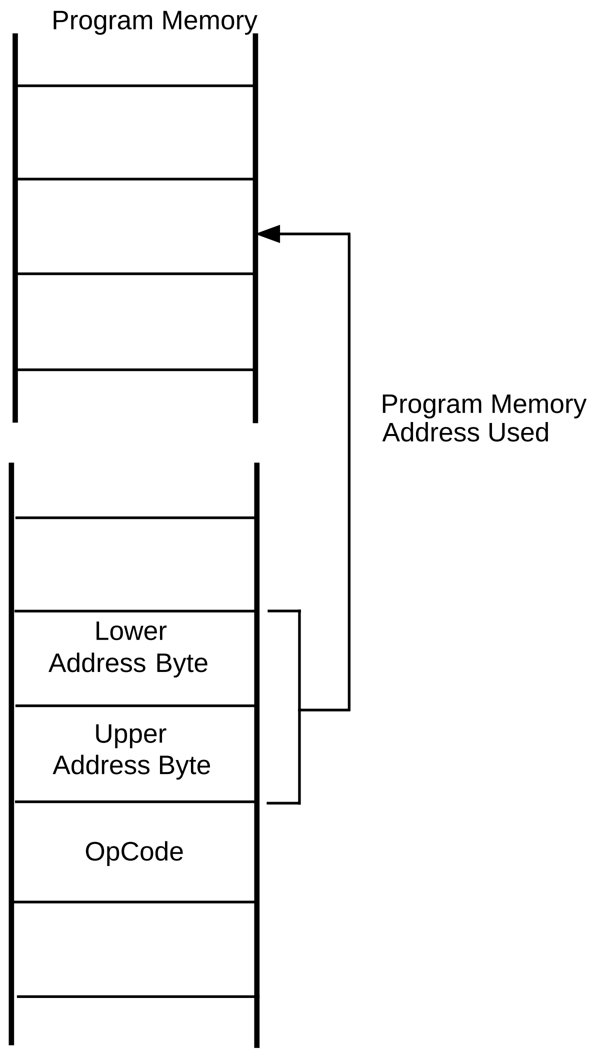


Figure 3-56. Direct Addressing

3.35 RELATIVE ADDRESSING

In the Relative Addressing (RA) Mode, illustrated in Figure 3-57, the instruction specifies a two's-complement signed displacement in the range of –128 to +127. This is added to the contents of the program counter (PC) to obtain the address of the next instruction to be executed.

The PC (prior to the add) consists of the address of the instruction following the Jump Relative (JR) or Decrement and Jump if Not Zero (DJNZ) instruction. JR and DJNZ are the only instructions which use this addressing mode.

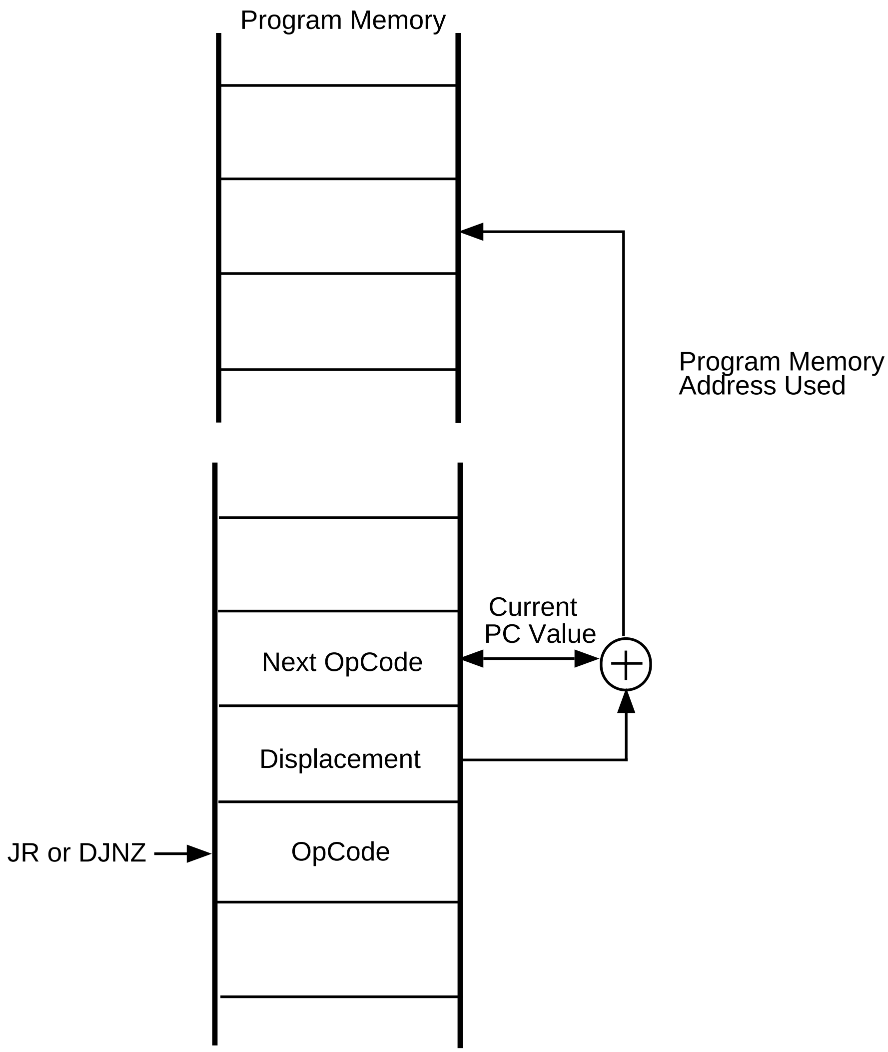


Figure 3-57. Relative Addressing

3.36 IMMEDIATE DATA ADDRESSING

Immediate (IM) Data is considered an addressing mode for the purposes of this discussion. It is the only addressing mode that does not indicate a register or memory address as the source operand.

The operand value used by the instruction is the value supplied in the operand field itself. Because an immediate operand is part of the instruction, it is always located in the program memory address space.

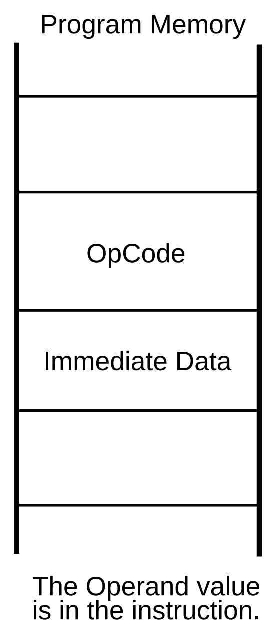


Figure 3-58. Immediate Data Addressing

3.37 INSTRUCTION SET FUNCTIONAL SUMMARY

Instructions can be divided functionally into the following eight groups:

- Load
- Bit Manipulation
- Arithmetic
- Block Transfer
- Logical
- Rotate and Shift
- Program Control
- CPU Control

The following tables show the instructions belonging to each instruction group and the number of operands required for each. The codes used for the operands are:

- src - Source Operand
- dst - Destination Operand
- cc - Condition Code

Table 3-15. Load Instructions

Mnemonic	Operands	Instruction
CLR	dst	Clear
LD	dst, src	Load
LDC	dst, src	Load Constant
LDE	dst, src	Load External
POP	dst	Pop
PUSH	src	Push

Table 3-16. Arithmetic Instructions

Mnemonic	Operands	Instruction
ADC	dst, src	Add with Carry
ADD	dst, src	Add
CP	dst, src	Compare
DA	dst	Decimal Adjust
DEC	dst	Decrement
DECW	dst	Decrement Word
INC	dst	Increment
INCW	dst	Increment Word
SBC	dst, src	Subtract with Carry
SUB	dst, src	Subtract

Table 3-17. Logical Instructions

Mnemonic	Operands	Instruction
AND	dst, src	Logical AND
COM	dst	Complement
OR	dst, src	Logical OR
XOR	dst, src	Logical Exclusive OR

Table 3-18. Program Control Instructions

Mnemonic	Operands	Instruction
CALL	dst	Call Procedure
DJNZ	dst, src	Decrement and Jump Non-Zero
IRET		Interrupt Return
JP	cc, dst	Jump
JR	cc, dst	Jump Relative
RET		Return

Table 3-19. Bit Manipulation Instructions

Mnemonic	Operands	Instruction
TCM	dst, src	Test Complement Under Mask
TM	dst, src	Test Under Mask
AND	dst, src	Bit Clear
OR	dst, src	Bit Set
XOR	dst, src	Bit Complement

Table 3-20. Block Transfer Instructions

Mnemonic	Operands	Instruction
LDCI	dst, src	Load Constant Auto Increment
LDEI	dst, src	Load External Auto Increment

Table 3-21. Rotate and Shift Instructions

Mnemonic	Operands	Instruction
RL	dst	Rotate Left
RLC	dst	Rotate Left Through Carry
RR	dst	Rotate Right
RRC	dst	Rotate Right Through Carry
SRA	dst	Shift Right Arithmetic
SWAP	dst	Swap Nibbles

Table 3-22. CPU Control Instructions

Mnemonic	Operands	Instruction
CCF		Complement Carry Flag
DI		Disable Interrupts
EI		Enable Interrupts
HALT		Halt
NOP		No Operation
RCF		Reset Carry Flag
SCF		Set Carry Flag
SRP	src	Set Register Pointer
STOP		Stop
WDH		WDT Enable During Halt
WDT		WDT Enable or Refresh

3.38 PROCESSOR FLAGS

The Flag Register (FCh) informs the user of the current status of the microcontroller. The flags and their bit positions in the Flag Register are shown in Figure 3-59.

The Flag Register contains six bits of status information which are set or cleared by CPU operations. Four of the bits (C, V, Z, and S) can be tested for use with conditional Jump instructions. Two flags (H and D) cannot be tested and are used for BCD arithmetic. The two remaining bits in the flag register (F1 and F2) are available to the user, but they must be set or cleared by instructions and are not usable with conditional Jumps.

As with bits in the other control registers, the flag register bits can be set or reset by instructions; however, only those instructions that do not affect the flags as an outcome of the execution should be used (Load Immediate).

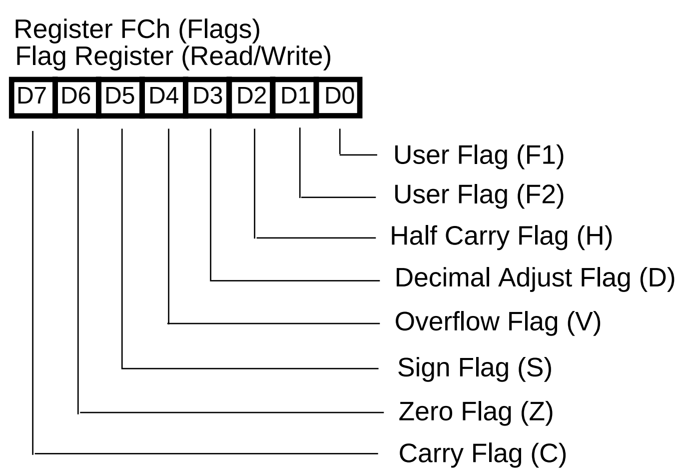


Figure 3-59. Flag Register

Note: The Watch-Dog Timer (WDT) instruction effects the Flags accordingly: Z=1, S=0, V=0.

3.38.1 Carry Flag

The Carry Flag (C) is set to 1 whenever the result of an arithmetic operation generates a carry out of or a borrow into the high-order bit 7. Otherwise, the carry flag is cleared to 0. An instruction can set, reset, or complement the carry flag. Following Rotate and Shift instructions, the carry flag contains the last value shifted out of the specified register. IRET may change the value of the carry flag when the Flag register, saved in the stack, is restored.

3.38.2 Zero Flag

For arithmetic and logical operations, the Zero Flag (Z) is set to 1 if the result is 0. Otherwise, the Zero Flag is cleared to 0. If the result of testing bits in a register is 00h, the Zero Flag is set to 1. Otherwise the Zero Flag is cleared to 0. If the result of a Rotate or Shift operation is 00h, the zero flag is set to 1. Otherwise, the Zero Flag is cleared to 0. IRET changes the value of the Zero Flag when the flag register saved in the stack is restored. The WDT instruction sets the Zero Flag to 1.

3.38.3 Sign Flag

The Sign Flag (S) stores the value of the most significant bit of a result following an arithmetic, logical, ROTATE, or SHIFT operation. When performing arithmetic operations on signed numbers, binary two's-complement notation is used to represent and process information. A positive number is identified by a 0 in the most significant bit position (bit 7); therefore, the Sign Flag is also 0. A negative number is identified by a 1 in the most significant bit position (bit 7); therefore, the Sign Flag is also 1. IRET changes the value of the Sign Flag when the flag register saved in the stack is restored.

3.38.4 Overflow Flag

For signed arithmetic, ROTATE, and SHIFT operations, the Overflow Flag (V) is set to 1 when the result is greater than the maximum possible number (>127) or less than the minimum possible number (<–128) that can be represented in two's-complement form. The Overflow Flag is set to 0 if no overflow occurs. Following logical operations the Overflow Flag is set to 0. IRET changes the value of the overflow flag when the flag register saved in the stack is restored.

3.38.5 Decimal-Adjust Flag

The Decimal-Adjust Flag (D) is used for BCD arithmetic. Since the algorithm for correcting BCD operations is different for addition and subtraction, this flag specifies what type of instruction was last executed so that the subsequent Decimal Adjust (DA) operation can function properly. Normally, the Decimal Adjust Flag cannot be used as a test condition. After a subtraction, the Decimal Adjust Flag is set to 1. Following an addition it is cleared to 0. IRET changes the value of the Decimal Adjust Flag when the flag register saved in the stack is restored.

3.38.6 Half-Carry Flag

The Half-Carry Flag (H) is set to 1 whenever an addition generates a carry out of bit 3 (overflow) or a subtraction generates a borrow into bit 3. The Half Carry Flag is used by the Decimal Adjust (DA) instruction to convert the binary result of a previous addition or subtraction into the correct decimal (BCD) result. As in the case of the Decimal Adjust Flag, the user does not normally access this flag. IRET changes the value of the Half Carry Flag when the flag register saved in the stack is restored.

3.39 CONDITION CODES

The C, Z, S, and V Flags control the operation of the Conditional Jump instructions. Sixteen frequently useful functions of the flag settings are encoded in a 4-bit field called the Condition Code (CC), which forms bits 4-7 of the conditional instructions.

Condition codes and flag settings are summarized in Table 3-23, Table 3-24, and Table 3-25.

Table 3-23. Flag Definitions

Flag	Description
C	Carry Flag
Z	Zero Flag
S	Sign Flag
V	Overflow Flag
D	Decimal Adjust Flag
H	Half Carry Flag

Table 3-24. Flag Settings Definitions

Symbol	Definition
0	Cleared to 0
1	Set to 1
*	Set or cleared according to operation
–	Unaffected
X	Undefined

Table 3-25. Condition Codes

Binary	HEX	Mnemonic	Definition	Flag Settings
0000	0	F	Always False	–
1000	8	(blank)	Always True	–
0111	7	C	Carry	C = 1
1111	F	NC	No Carry	C = 0
0110	6	Z	Zero	Z = 1
1110	E	NZ	Non-Zero	Z = 0
1101	D	PL	Plus	S = 0
0101	5	MI	Minus	S = 1
0100	4	OV	Overflow	V = 1
1100	C	NOV	No Overflow	V = 0

Table 3-25. Condition Codes (Continued)

Binary	HEX	Mnemonic	Definition	Flag Settings
0110	6	EQ	Equal	Z = 1
1110	E	NE	Not Equal	Z = 0
1001	9	GE	Greater Than or Equal	(S XOR V) = 0
0001	1	LT	Less Than	(S XOR V) = 1
1010	A	GT	Greater Than	(Z OR (S XOR V)) = 0
0010	2	LE	Less Than or Equal	(Z OR (S XOR V)) = 1
1111	F	UGE	Unsigned Greater Than or Equal	C = 0
0111	7	ULT	Unsigned Less Than	C = 1
1011	B	UGT	Unsigned Greater Than	(C = 0 AND Z = 0) = 1
0011	3	ULE	Unsigned Less Than or Equal	(C OR Z) = 1

3.40 NOTATION AND BINARY ENCODING

In the detailed instruction descriptions that make up the rest of this chapter, operands and status flags are represented by a notational shorthand. Operands, condition codes, address modes, and their notations are as follows:

Table 3-26. Notational Shorthand

Notation	Address Mode	Operand	Range[1]
cc	Condition Code		See condition codes
r	Working Register	Rn	n = 0 – 15
R	Register or Working Register	Reg Rn	Reg represents a number in the range of 00h to FFh n = 0 – 15
RR	Register Pair or Working Register Pair	Reg RRp	Reg represents an even number in the range of 00h to FEh p = 0, 2, 4, 6, 8, 10, 12, or 14
Ir	Indirect Working Register	@Rn	n = 0 –15
IR	Indirect Register	@Reg	Reg represents a number in the range of 00h to FFh

Table 3-26. Notational Shorthand (Continued)

Notation	Address Mode	Operand	Range[1]
	or		
	Indirect Working Register	@Rn	n = 0 – 15
Irr	Indirect Working Register Pair	@RRp	p = 0, 2, 4, 6, 8, 10, 12, or 14
IRR	Indirect Register Pair	@Reg	Reg represents an even number in the range 00h to FFh
	or		
	Working Register Pair	@RRp	p = 0, 2, 4, 6, 8, 10, 12, or 14
X	Indexed	Reg (Rn)	Reg represents a number in the range of 00h to FFh and n = 0 – 15
DA	Direct Address	Addrs	Addrs represents a number in the range of 00h to FFh
RA	Relative Address	Addrs	Addrs represents a number in the range of +127 to –128 which is an offset relative to the address of the next instruction
IM	Immediate	#Data	Data is a number between 00h to FFh

Note:

1. See the device product specification to determine the exact register file range available. The register file size varies by the device type.

Additional notation includes:

Table 3-27. Additional Notation

Symbol	Definition
dst	Destination Operand
src	Source Operand
@	Indirect Address Prefix
SP	Stack Pointer
PC	Program Counter
FLAGS	Flag Register (FCh)
RP	Register Pointer (FDh)
IMR	Interrupt Mask Register (FBh)
#	Immediate Operand Prefix
%	Hexadecimal Number Prefix
h	Hexadecimal Number Suffix
b	Binary Number Suffix
OPC	Opcode

3.40.1 Assembly Language Syntax

For proper instruction execution, assembly language syntax requires ‘dst, src’ be specified in that order. The following instruction descriptions show the format of the object code produced by the assembler. This binary format should be followed by users who prefer manual program coding or who intend to implement their own assembler.

Example: The contents of registers 43h and 08h are added and the result is stored in 43h. The assembly syntax and resulting object code are:

ASM:	ADD	43h,	08h	(ADD dst, src)
OBJ:	04	08	43	(OPC src, dst)

CHAPTER 4
ON-SCREEN DISPLAY

4.1 INTRODUCTION

The On-Screen Display (OSD) generates and displays a 10 row by 24 columns of 256 characters at 14- x 18-dots resolution. The color of each character is specified on a row basis.

The DTC detects H_{SYNC} and V_{SYNC} signals to synchronize its internal circuitry to the video signal, then outputs RGB and Video Blank (VBLANK) signals. The VBLANK signal is used to multiplex the OSD signal and video signal onto the screen. The result is that the On-Screen Display is superimposed over the TV picture.

The display results from the successful timing of several components:

- OSD Positioning
- Second Color Feature
- Mesh and Halftone Effect
- OSD Fade
- Inter-Row Spacing
- Character Generation

The OSD format is 10 rows containing 24 columns. Row and column numbering begin with the number 0. A full row contains 24 characters which can be referred to as columns 0 through 23. The 10 rows of the OSD can be referred to as rows 0 through 9.

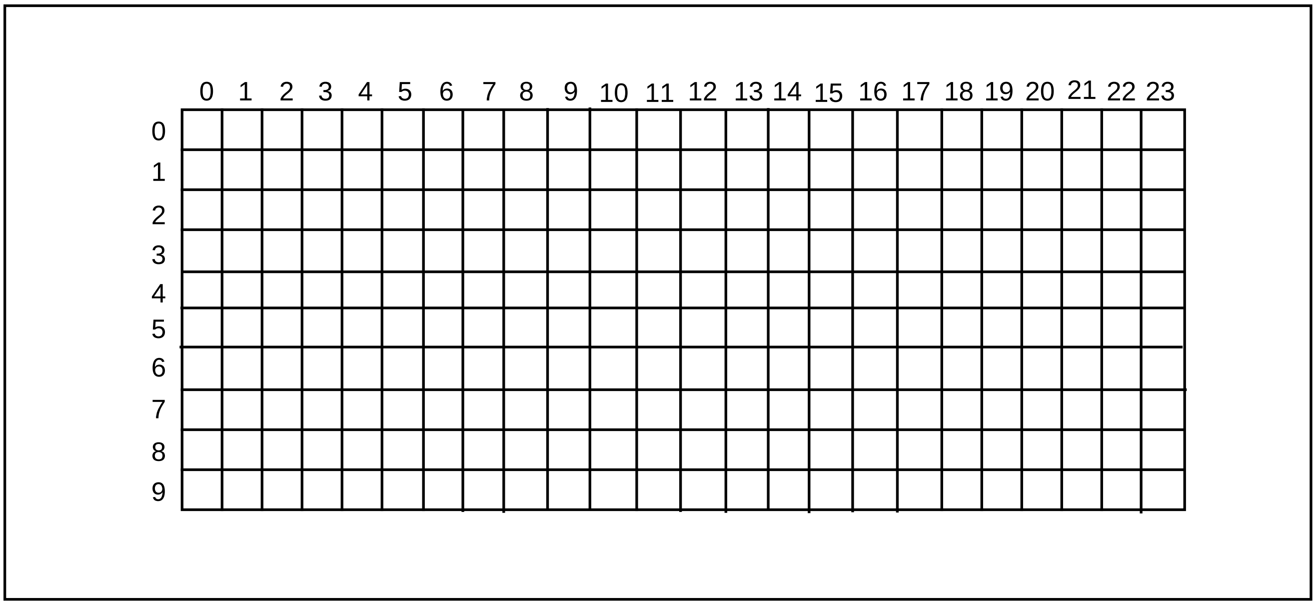


Figure 4-1. OSD Format

4.2 OSD POSITION

OSD Positioning is controlled by programming the following registers:

- OSD Control Register
- Vertical Position Register
- Horizontal Position Register

4.2.1 OSD Control Register

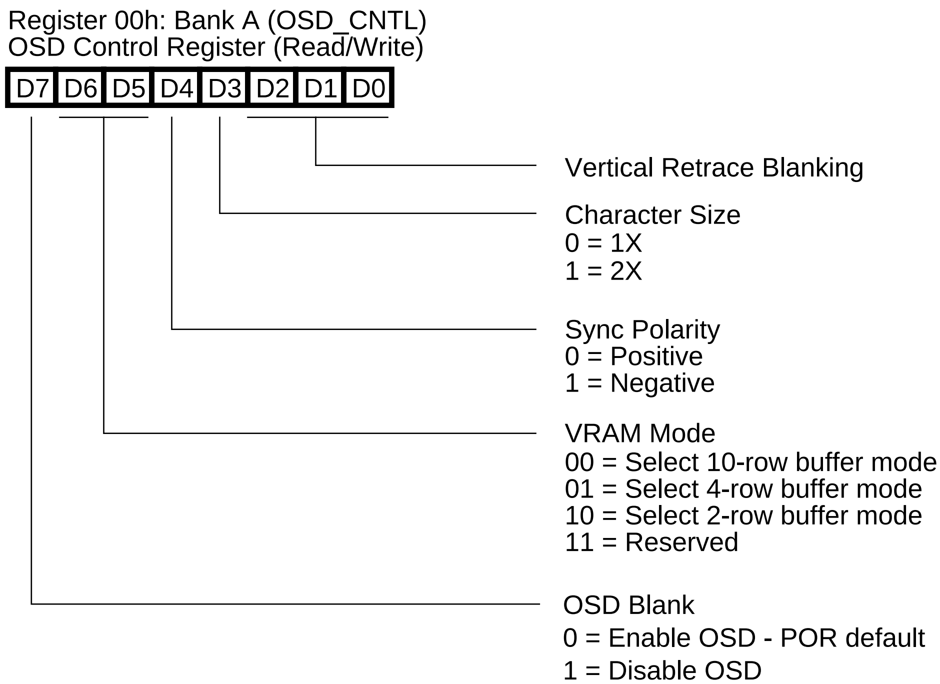


Figure 4-2. OSD Control Register

Bit 7, OSD Blank, enables or disables the OSD. When the value is set to 0, the OSD is available for use. When the value is set to 1, the OSD is disabled.

Bits 6 and 5, VRAM Mode select 10-, 4-, or 2-row buffer mode.

Bit 4, Sync Polarity, provides the polarity of the H_{SYNC} and V_{SYNC} signals. H_{SYNC} and V_{SYNC} must have the same polarity. This feature is designed to provide flexibility for TV chassis designers.

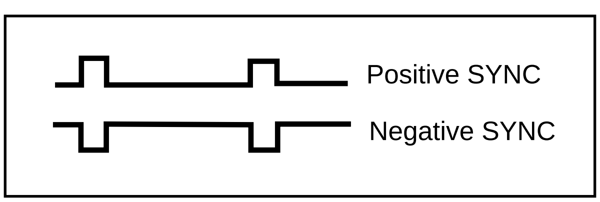


Figure 4-3. Positive and Negative Sync Signals

Bit 3, Character Size, sets the size of the characters that are displayed. Two sizes are supported—1X and 2X. The default value is 1X. To change the size of the characters in a row, alter the value of the bit during the previous horizontal interrupt. The character size of the first row is programmed during vertical interrupt (V_{SYNC}) processing. Character size is a row interrupt-driven attribute.

Bits 2, 1, and 0, Vertical Retrace Blanking, sets a time period when the OSD is disabled while the electron gun returns from the bottom to the top of the screen, and all VBLANK and RGB output are disabled. The blanking period is determined by counting horizontal pulses as follows:

Blanking Period=(4 x (Vertical Retrace Blanking)+2) x THL

The retrace blanking bits, OSD_CNTL (2,1,0) must be set to deactivate the electron guns during the retrace period. During vertical retrace, no video information is available in the TV signal for display. OSD should not be displayed at every retrace, so it must be blanked out.

4.2.2 Vertical Position Register

The Vertical Position Register sets the vertical placement of the OSD on the screen. The unit of measure for placement is the number of scan lines from the top of the display screen.

Bits 7 and 6 are reserved for future use. If this register is read, these bits return 1s.

Bits 5, 4, 3, 2, 1, and 0, Vertical Position, specify the vertical position of the OSD window from the start of V_{SYNC}.

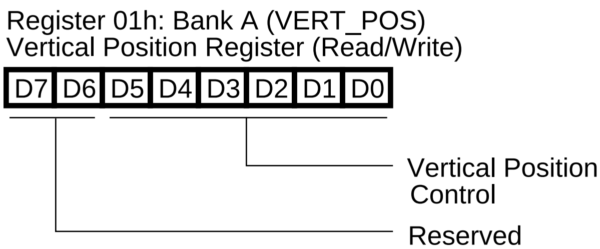


Figure 4-4. Vertical Position Register

The value required for this register may be computed using the following equation:

$$VERT\_POS = (V_{POS} - 6) / 4$$

VERT_POS represents the contents of bits 5,4,3,2,1,0 of the Vertical Position Register (VERT_POS). The default value is 0. When the value is 0, the OSD is at the top-most OSD position on the screen, with an offset of 06h scan lines above the OSD area.

V_{POS} is the number of scan lines from the V_{SYNC} to the OSD start position. V_{POS} must be a positive integer with a minimum value of Ah incrementing by 4. Some possible values include: 10, 14, 18, 22, 26, 30.

For example, V_{POS} = 22:

$$VERT\_POS = (22 - 6) / 4$$

$$VERT\_POS = 16 / 4$$

$$VERT\_POS = 4$$

The contents of the register VERT_POS (5,4,3,2,1,0) should be, for this example, set to:

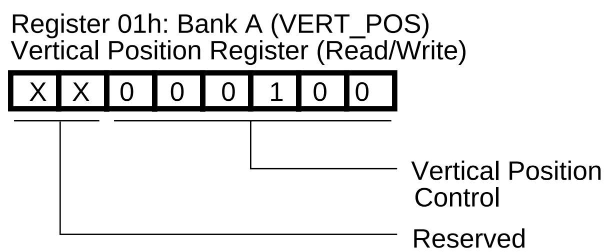


Figure 4-5. Vertical Position Example

4.2.3 Horizontal Position Register

The Horizontal Position Register sets the horizontal start position of the OSD. The unit of measure for placement is the number of pixels from the left of the display screen.

Bits 7 and 6 are reserved for future use. If this register is read, these bits return 1s.

Bits 5, 4, 3, 2, 1, and 0, Horizontal Position, specify the horizontal position of the OSD window from the start of H_{SYNC}.

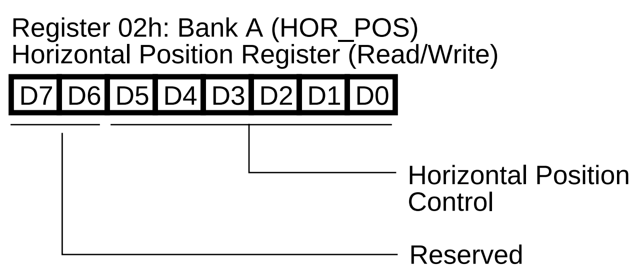


Figure 4-6. Horizontal Position Register

The value required for this register may be computed using the following equation:

$$HOR\_POS = (H_{POS} - 1) / 4$$

HOR_POS represents the contents of bits 5,4,3,2,1,0 of the Horizontal Position Register (HOR_POS). The default value is 0. When the value is 0, the OSD is at the left-most OSD position on the screen.

H_{POS} is the number of pixels from the left of the screen to the OSD start position. H_{POS} must be a positive integer with a minimum value of 5 incrementing by 4. Some possible values include: 9, 13, 17, 21, 25, 29.

For example, H_{POS} = 17:

$$HOR\_POS = (17 - 1) / 4$$

$$HOR\_POS = 16 / 4$$

$$HOR\_POS = 4$$

The contents of the register HOR_POS (5,4,3,2,1,0) should be, for this example, set to:

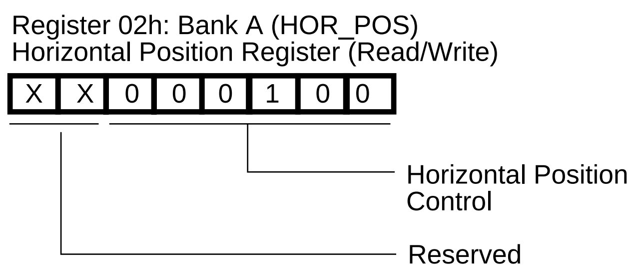


Figure 4-7. Horizontal Position Example

4.2.4 Second Color Feature

Second Color feature is the logical division of each column into two parts along each row for changing foreground color. The number of each half-column is called the Second Color Position.

The Second Color feature can be used for the smooth change of color in a row.

The change step for color is half of the character size.

4.2.5 Second Color Control Register

The Second Color Position is the place where the foreground color changes to the color defined in the Second Color Control Register.

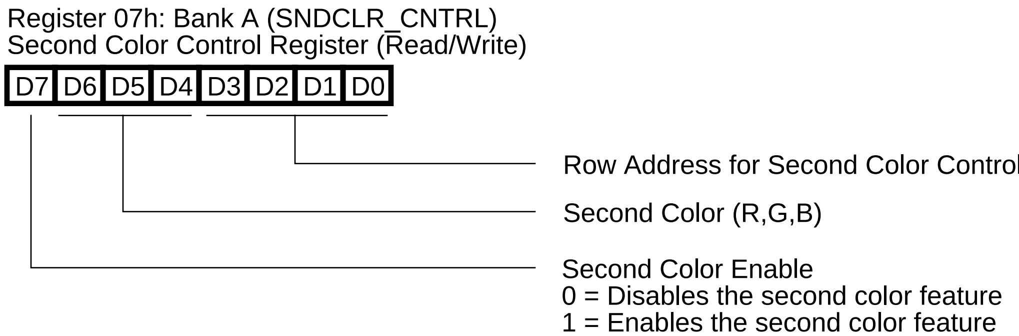


Figure 4-8. Second Color Control Register

4.2.6 Second Color Register

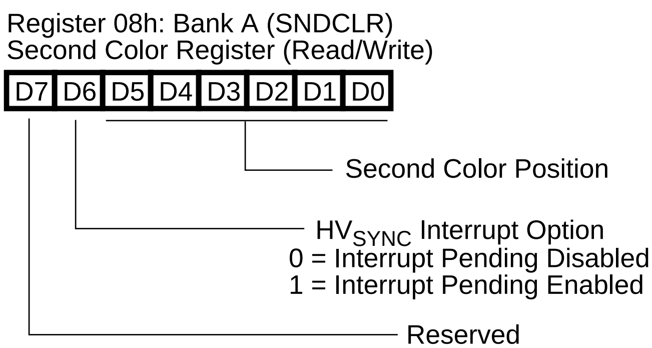


Figure 4-9. Second Color Register

Bit 7 is reserved. When the register is read, bit 7 returns a value of 1.

Bit 6, HV$_{SYNC}$ Interrupt Option, sets the procedure for processing when a second interrupt is issued before the processing of the first interrupt has completed. For example, an H$_{SYNC}$ interrupt comes in before completion of the V$_{SYNC}$ interrupt processing. If bit 6 is reset to 0 and interrupt request is disabled during vertical interrupt, the horizontal interrupt will be missed.

If bit 6 is set to 1 and the interrupt request is reset to 0 during vertical interrupt service, then the horizontal interrupt will be pended and serviced after completion of the vertical interrupt processing.

Bits 5, 4, 3, 2, 1, and 0, Second Color Position, control second color display. This field specifies the place at which to start the second color. A specific color can be assigned as the second color.

4.2.7 Second Color Example

Figure 4-10 illustrates a second color display in row 8 of the OSD. Each of the small-grid squares represents one pixel. Each column is comprised of two parts.

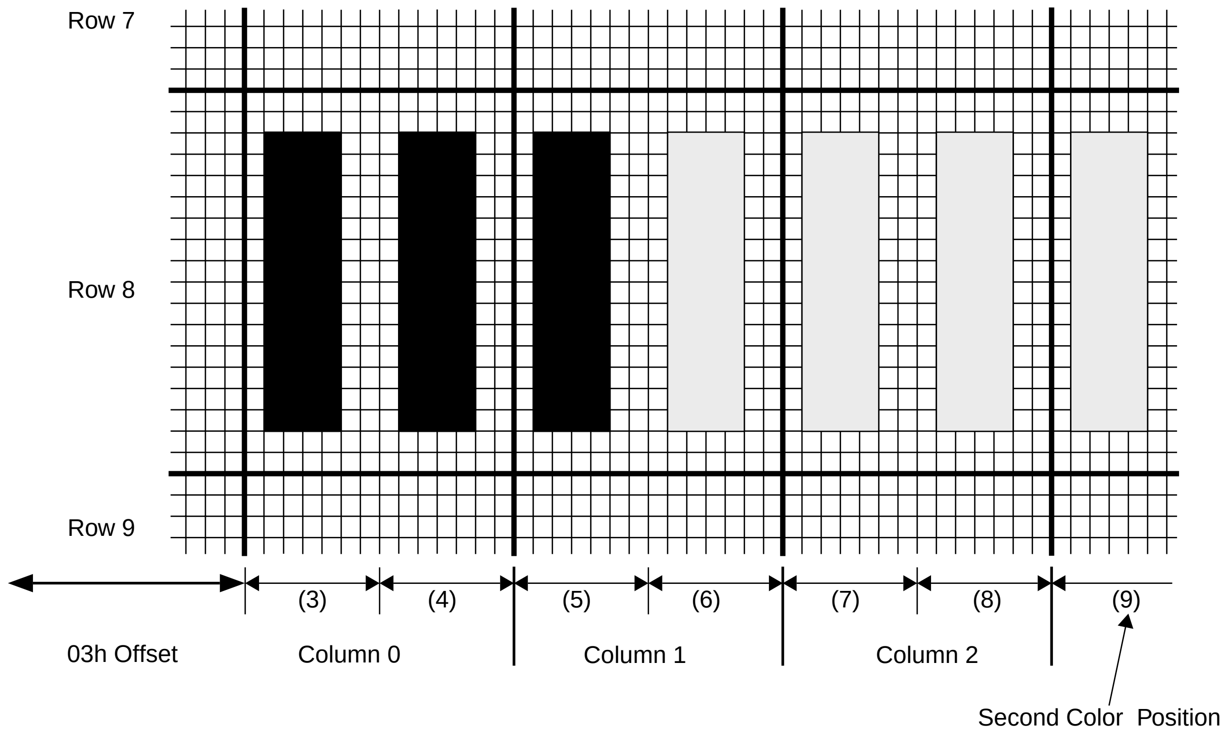


Figure 4-10. Second Color Example

In this figure, a second color is displayed at Second Color Position 6. The second color position for the first column has a value of 3 because the OSD is offset from the left of the TV screen a distance equal to 03h. Each column is the size of one display character. Each Second Color column is a half column, which is the same as a half character. The screen position offset is added to the Second Color Position.

In the example, the offset is 03h. Therefore, Second Color Positions begin with 3 = (3+0), 4 = (3+1), 5 = (3+2), and so forth. The change in color occurs at Second Color Position 6.

Before displaying row 8, the value of SNDCLR_CNTRL must be programmed as 11001000B, and the value of SNDCLR is XX000110B. The register values are illustrated in Figure 4-11.

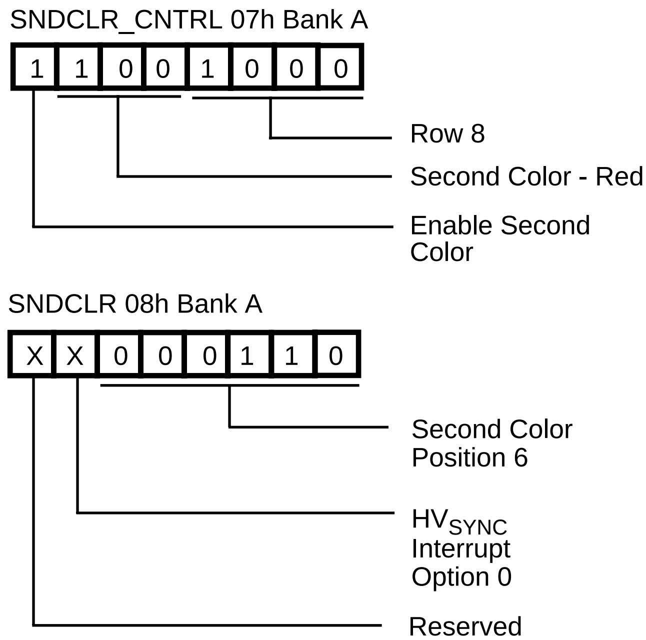


Figure 4-11. Second Color Example Registers

4.3 MESH AND HALFTONE EFFECT

Mesh is a grid-like area that contains alternating pixel display of OSD and transparent zones. The transparent zones allow the TV signal display to appear in part while the mesh display is active.

Halftone effect is a transparent area that appears slightly darker than the regular picture that is carried by the TV signal.

Mesh and halftone effects both serve as backgrounds for menus, action bars, and other On-Screen Displays. The mesh feature is only for interlaced-mode video systems.

Mesh can be controlled in two ways—through hardware or through software for alternating pixel display in different fields.

Software must define a character-based window in OSD to support mesh/halftone effects. The following control registers must be programmed properly to support the character-based mesh/halftone window:

- MC_St
- MC_End
- MR_En
- MC_Reg

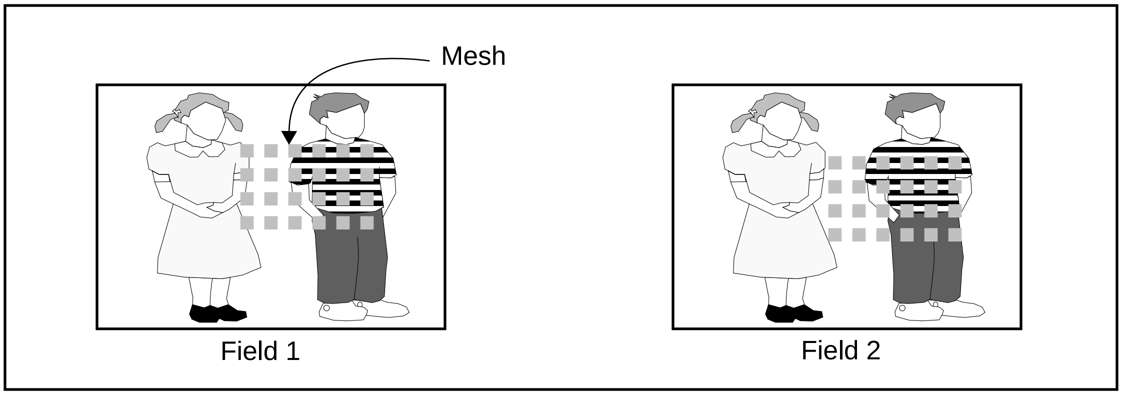


Figure 4-12. Mesh (Example)

A close-up example shows more precisely how the OSD overlays the TV signal when the mesh is active.

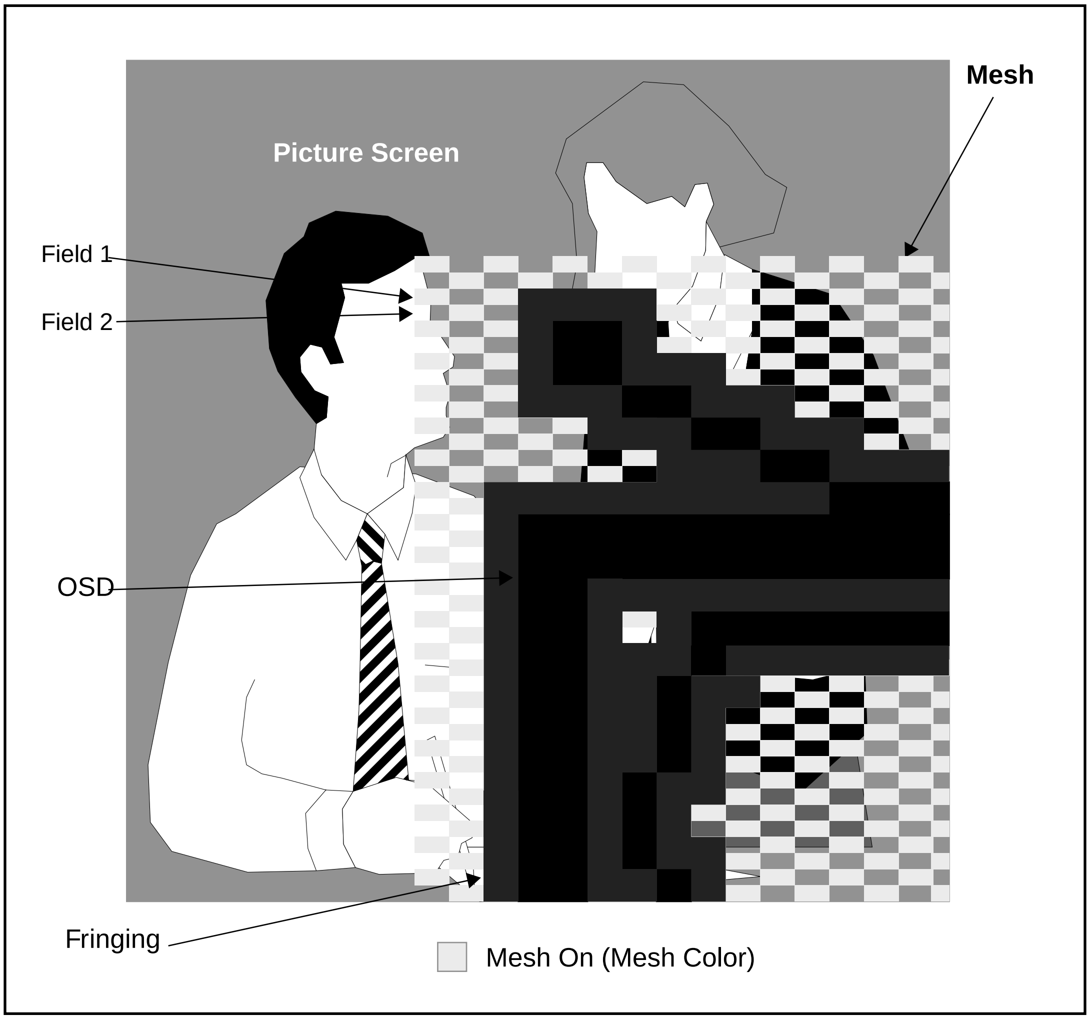


Figure 4-13. Mesh On

General descriptions of the registers used to control the mesh are contained in the tables below. An example appears after the tables to further describe this feature.

4.3.1 Mesh Column Start Register

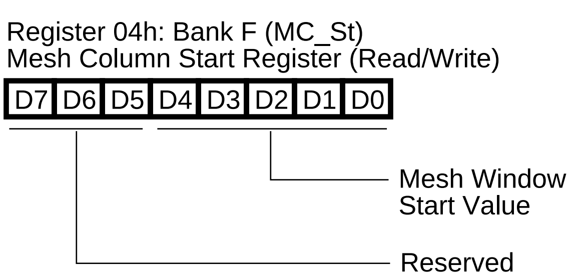


Figure 4-14. Mesh Column Start Register

4.3.2 Mesh Column End Register

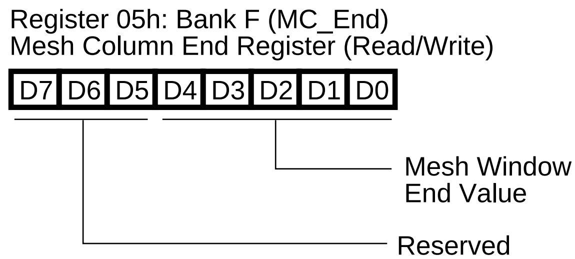


Figure 4-15. Mesh Column End Register

MC_St and MC_End define the width and horizontal position of the mesh window.

4.3.3 Mesh Row Enable Register

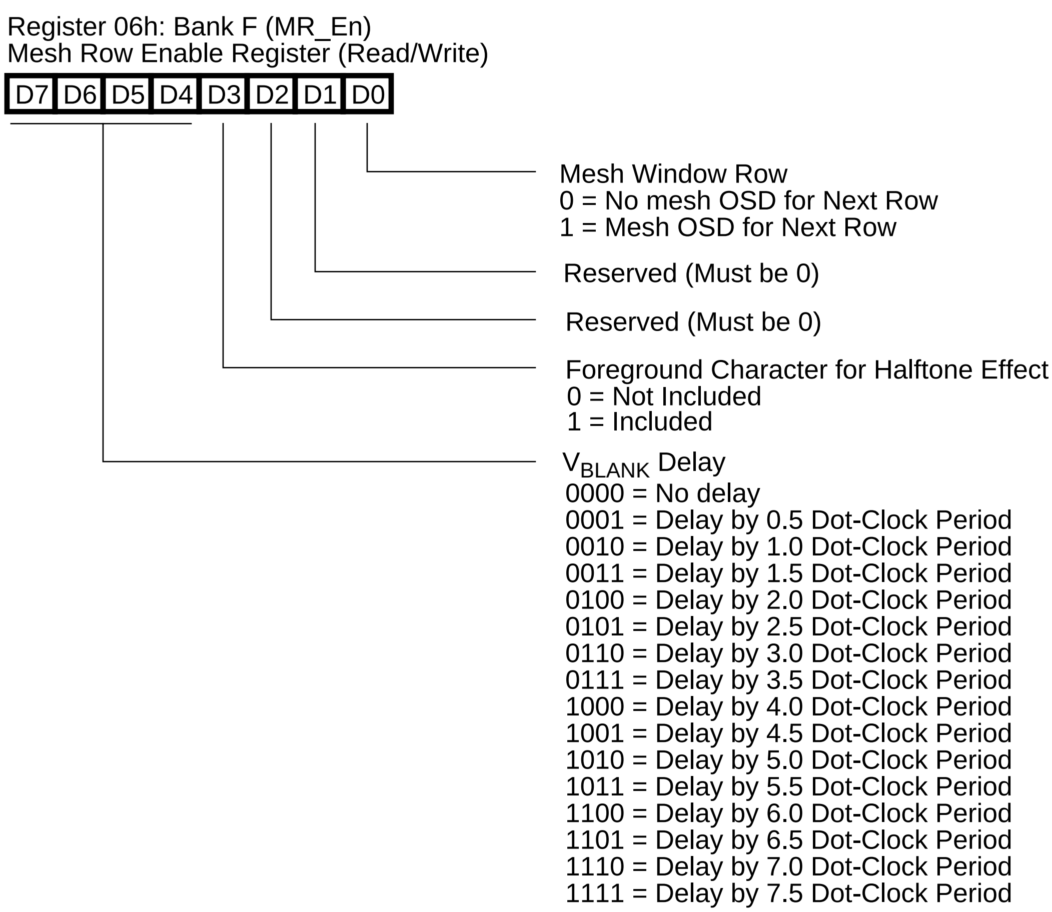


Figure 4-16. Mesh Row Enable Register

Bits 7, 6, 5, and 4, VBLANK Delay, is the amount of time that the VBLANK signal is properly aligned with the OSD RGB output with delay from external circuitries.

Bit 3, Character Foreground for Halftone Effect, defines whether display of a foreground color for character display is included. If bit 3 is set to 0, halftone is disabled for pixels with foreground color. If bit 3 is set to 1, halftone is active for pixels with both foreground and background colors.

Bit 2, Character Background Color with Halftone Effect on P20, is Reserved, and must be 0.

Bit 1, Character Background Display Enable, is Reserved, and must be 0.

Bit 0, Mesh Window Row, sets the mesh window to On or Off for the next row of the OSD.

4.3.4 Mesh Control Register

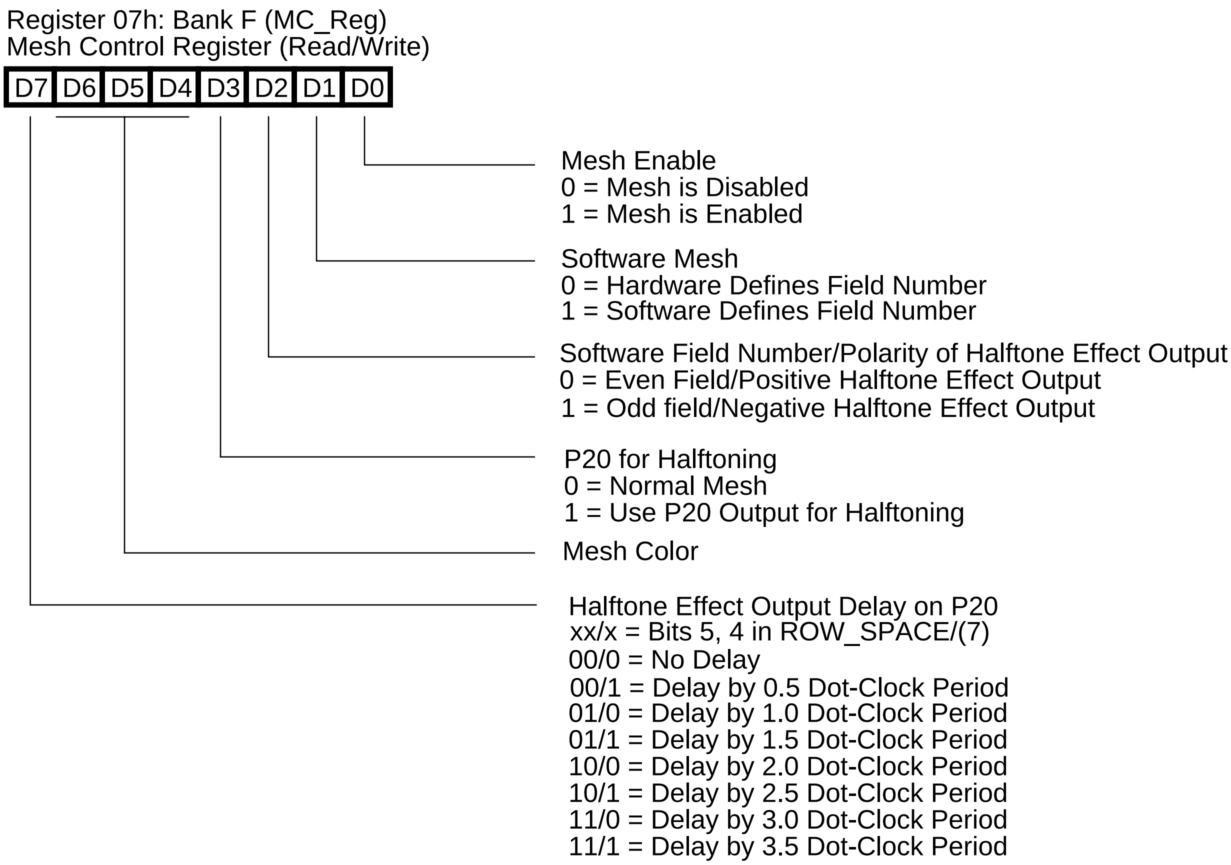


Figure 4-17. Mesh Control Register

Note: The order of the colors differs from the order (Red, Green, Blue) of the Second Color field of the SNDCLR_CNTRL register.

Bit 7, Halftone Output Delay on P20, is the amount of time that output of the halftone signal is delayed to compensate for the amount of delay of OSD RGB from external circuitries.

Bits 6, 5, and 4, Mesh Color, defines the color of the mesh window. The colors are specified in Blue, Green, Red order, as shown in Table 4-1.

Table 4-1. BGR Mesh Colors

B	G	R	Color
0	0	0	Black
0	0	1	Red
0	1	0	Green
0	1	1	Yellow
1	0	0	Blue
1	0	1	Magenta
1	1	0	Cyan
1	1	1	White

Bit 3, P20 for Halftone, selects mesh or halftone effect. If bit 3 is set to 1, P20 outputs halftone. If reset to 0, P20 is a normal I/O pin.

Bit 2, Software Field Number/Polarity of Halftone Output, has several possible values. The value of this bit remains the same for the entire mesh window; it does not change from row to row.

If bit 3 is set to 1 (halftone), bit 2 defines the polarity of halftone output. If bit 3 is reset to 0 and bit 1 is set to 1, then bit 2 defines the field number (even or odd).

Bit 1, Software Mesh, sets whether hardware or software defines the current field number. When the value equals 0, hardware defines field number. When the value equals 1, software defines the field number.

Bit 0, Mesh Enable, disables or enables use of the mesh. This field is used in conjunction with MR_EN (0). The value of Mesh Enable should be changed only when Mesh Window Row equals 0 (the current OSD row is not part of a mesh window). If the value is changed when the current row is part of the mesh window, partial or missing characters are likely to be displayed.

4.3.5 Mesh Window Display Example

A software-controlled mesh window is to be displayed in columns 2 through 20 of rows 3 through 7 of the OSD.

At the start of the display of the OSD (row 0), the values in the registers are as follows:

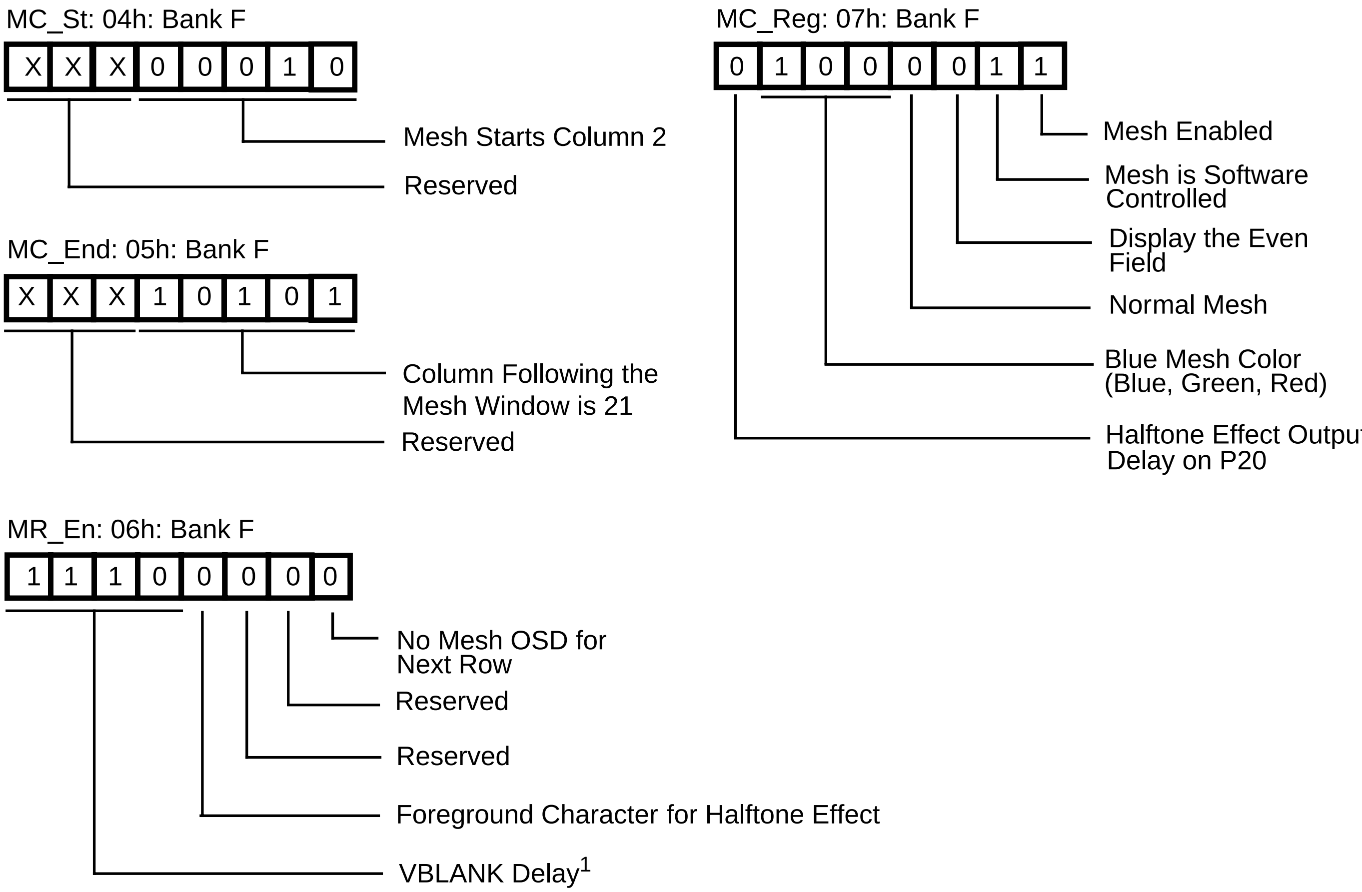


Figure 4-18. Mesh Window Display Registers for Row 0 (Example)

Note:

1. The value shown for VBLANK Delay is not significant. For this example, the value is unused; bits 7-4 would equal some previously assigned value.

When the H_{SYNC} interrupt is issued to start the display of Row 2, register values are the same as for Row 0 with one exception—MR_En (0) would be 1, rather than 0. Mesh Window Row must indicate that the following row, Row 3 is to be included in the mesh window. When the Row 1 interrupt is issued, the registers have the following values:

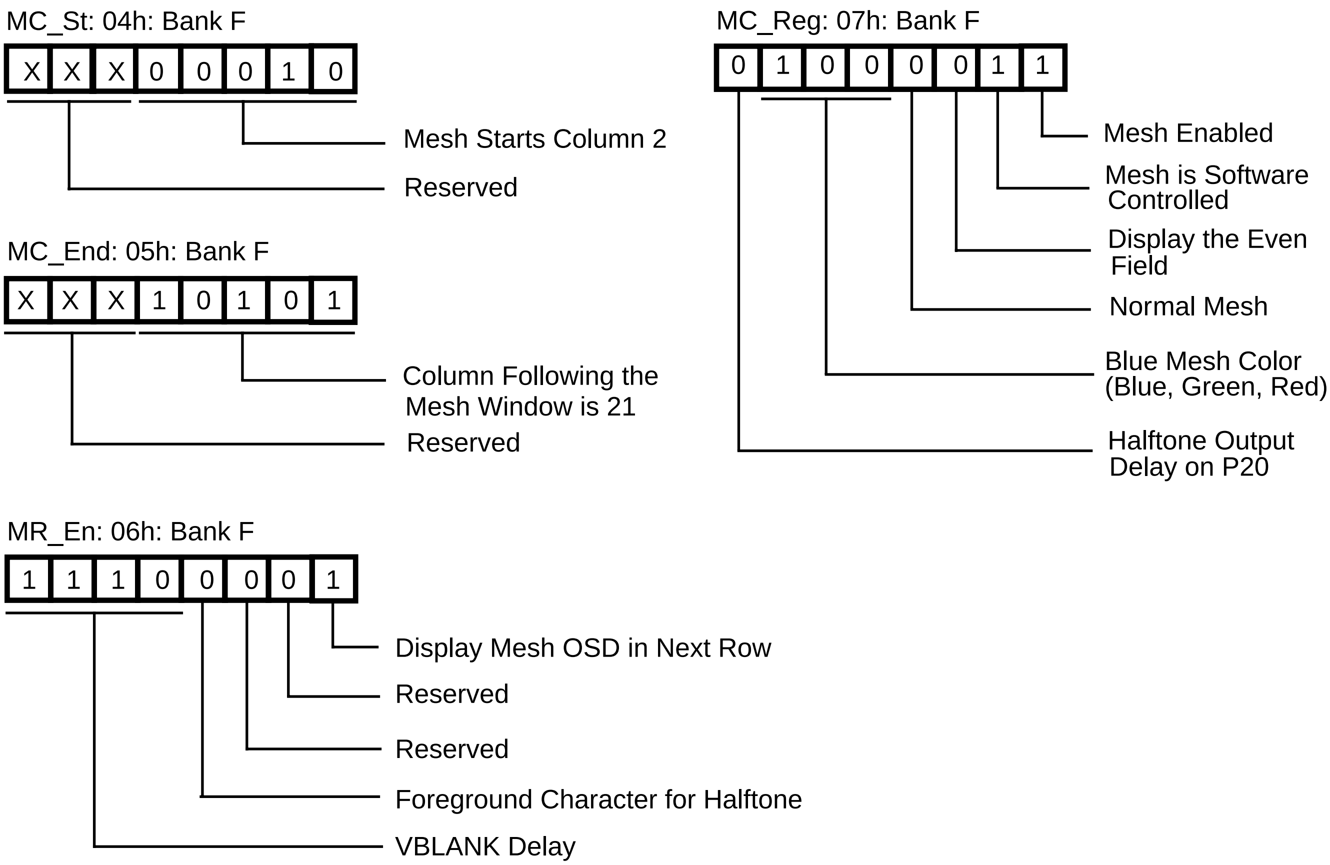


Figure 4-19. Mesh Window Display Registers for Row 1-6 (Example)

When the interrupt is issued to start the display of Row 2, these registers have exactly the same values as shown in Figure 4-19; the values are unchanged from the start of Row 1.

In fact, the values remain the same until prior to the display of row 7, when the Mesh Window Row value reverts to 0, indicating that Row 8 is not included in the mesh window.

Mesh and halftone effects are configured identically with the exception of bit D3 on expanded Register Bank F (MC_REG). For halftone effect, set bit D3 to 1. For mesh, set bit D3 to 0.

Note: Port 2 must be configured to output for halftone effect.

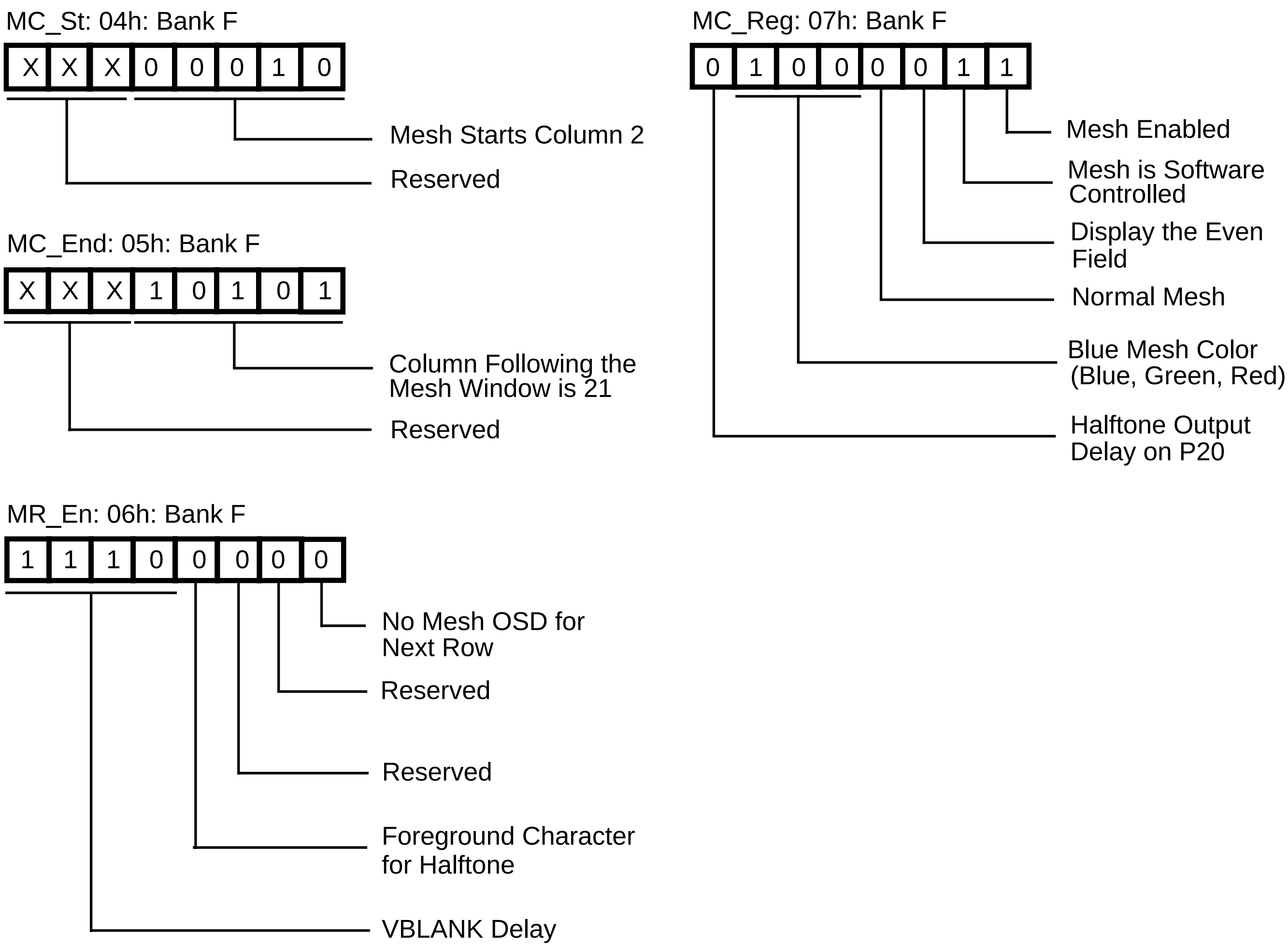


Figure 4-20. Mesh Window Display Registers for Row 7 (Example)

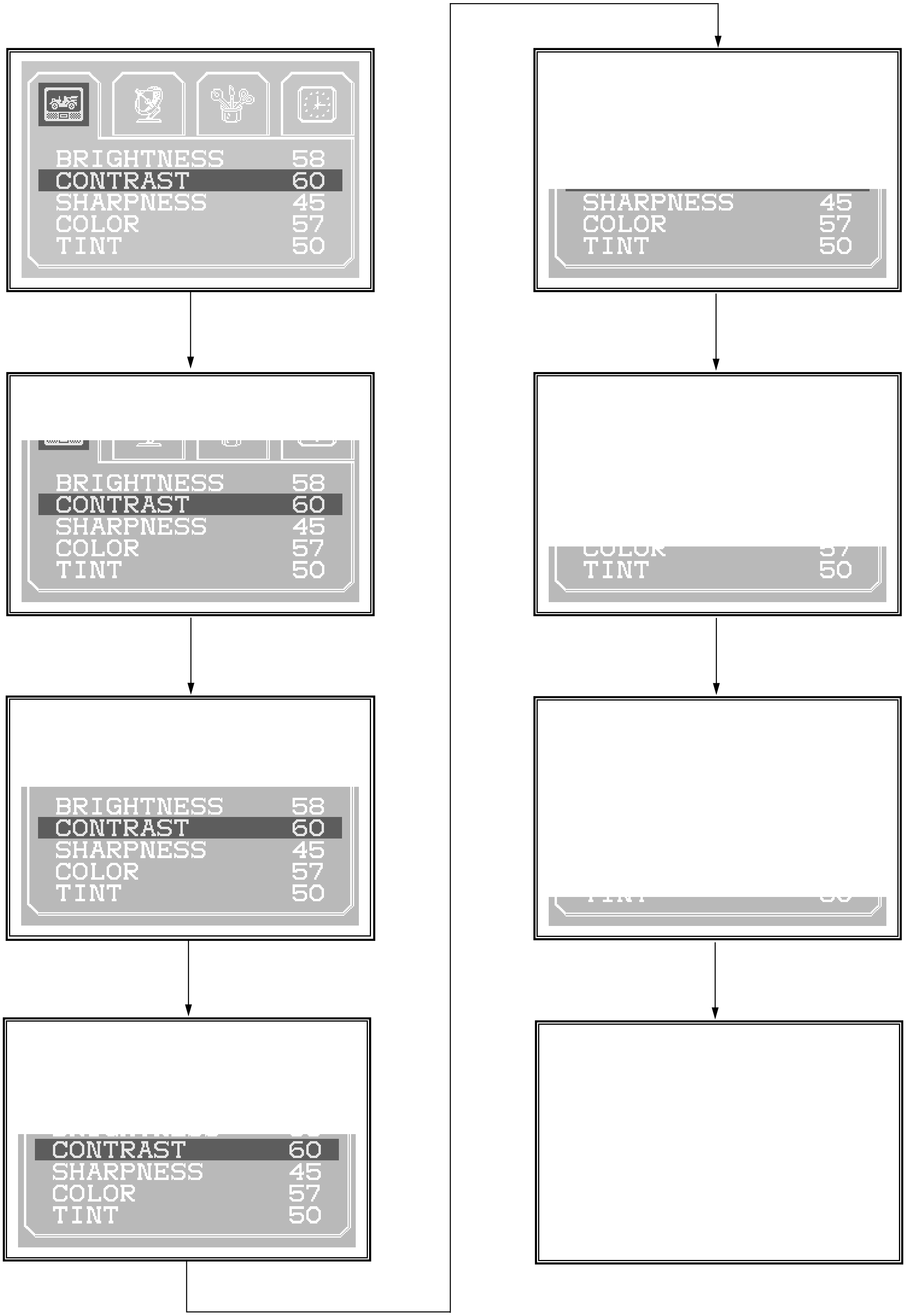
The values of these registers would remain unchanged for the remaining rows of the field.

4.4 OSD FADE

Fading is the gradual disappearance of the OSD. Fading occurs vertically, up or down. Figure 4-21 demonstrates the fade-down effect.

Fade control registers must be updated only during V_{SYNC}, not during row interrupt. Otherwise, unexpected results might occur.

Figure 4-21. Video Fade (Example)


BRIGHTNESS 58
CONTRAST 60
SHARPNESS 45
COLOR 57
TINT 50
BRIGHTNESS 58
CONTRAST 60
SHARPNESS 45
COLOR 57
TINT 50
BRIGHTNESS 58
CONTRAST 60
SHARPNESS 45
COLOR 57
TINT 50
CONTRAST 60
SHARPNESS 45
COLOR 57
TINT 50
SHARPNESS 45
COLOR 57
TINT 50
TINT 50

This feature is controlled through the FADE_POS1, FADE_POS2, and ROW_SPACE registers.

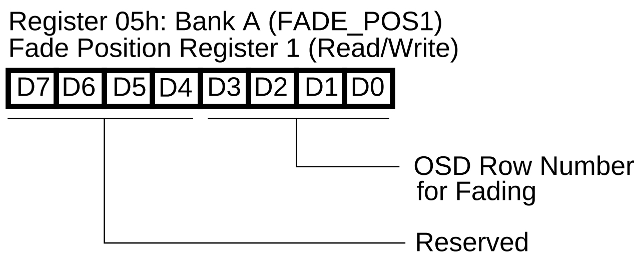


Figure 4-22. Fade Position Register 1

Bits 3, 2, 1, and 0 defines the boundary row for the fade area. The portion of the OSD above or below the row number fades up or down, as set in Fade Direction, ROW_SPACE (6).

The fade starts at the scan line set in FADE_POS2 (4,3,2,1,0) within the row number set in FADE_POS1 (3,2,1,0).

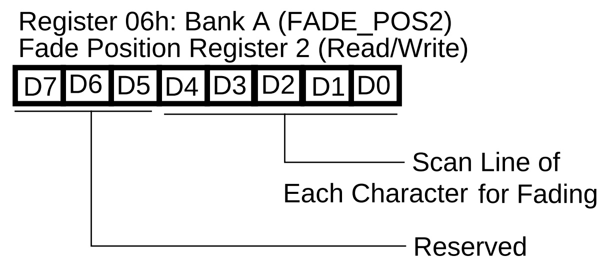


Figure 4-23. Fade Position Register 2

4.5 INTER-ROW SPACING

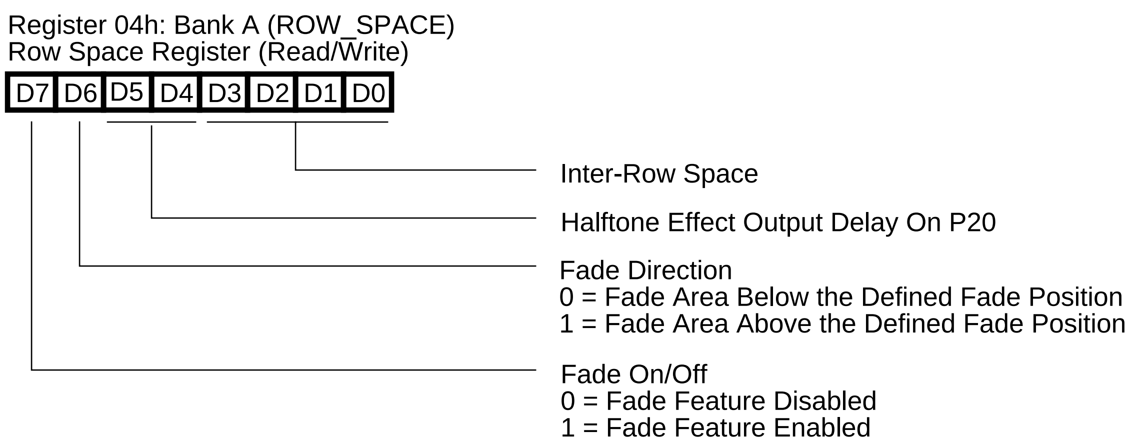


Figure 4-24. Row Space Register

Bit 7, Fade On/Off, disables or enables the fade effect. When Fade On/Off is reset to 0, the entire OSD is displayed. When Fade On/Off is set to 1, a portion of the OSD is transparent.

Bit 6, Fade Direction, controls the direction the fade appears to move on the screen. When Fade Direction is set to 0, fading moves toward the bottom of the TV screen. Fading occurs beginning with the row number set in FADE_POS1 (3,2,1,0) and the scan line number set in FADE_POS2 (4,3,2,1,0). For example, fading could begin in row 0 scanline 0 and move down the screen. When the Fade Direction is set to 1, fading is toward the top of the screen. Fading occurs beginning with the row number set in FADE_POS1 (3,2,1,0) and the scan line number set in FADE_POS2 (4,3,2,1,0). For example, fading could begin in row 9 scanline 17 and move up the screen.

Bits 5 and 4, Halftone Effect Delay on P20, works with MC_REG (7).

Bits 3, 2, 1, and 0, Inter-Row Space, specifies a number of Horizontal Scan Lines (HL) to add between displayed rows.

Inter-Row Spacing can be from 0 to 15 HL. A setting of 0 HL is called Continuous Row Display.

The spacing between any two rows can be controlled by programming it during the period of the previous horizontal interrupt service. A horizontal interrupt is generated at the start of each character row. Software must program the spacing between the current row and the next row during the current horizontal interrupt.

The amount of time required to process a row should not exceed the display time of the row.

4.6 CHARACTER GENERATION

Character generation provides the content of the OSD. The Z90230-family of products support a true 14-pixel (horizontal) by 18-pixel (vertical) character display with 256 character sets.

4.6.1 Character Cell Resolution

To achieve improved performance, characters are mapped pixel-by-pixel in Character Generation Read-Only Memory (CGROM).

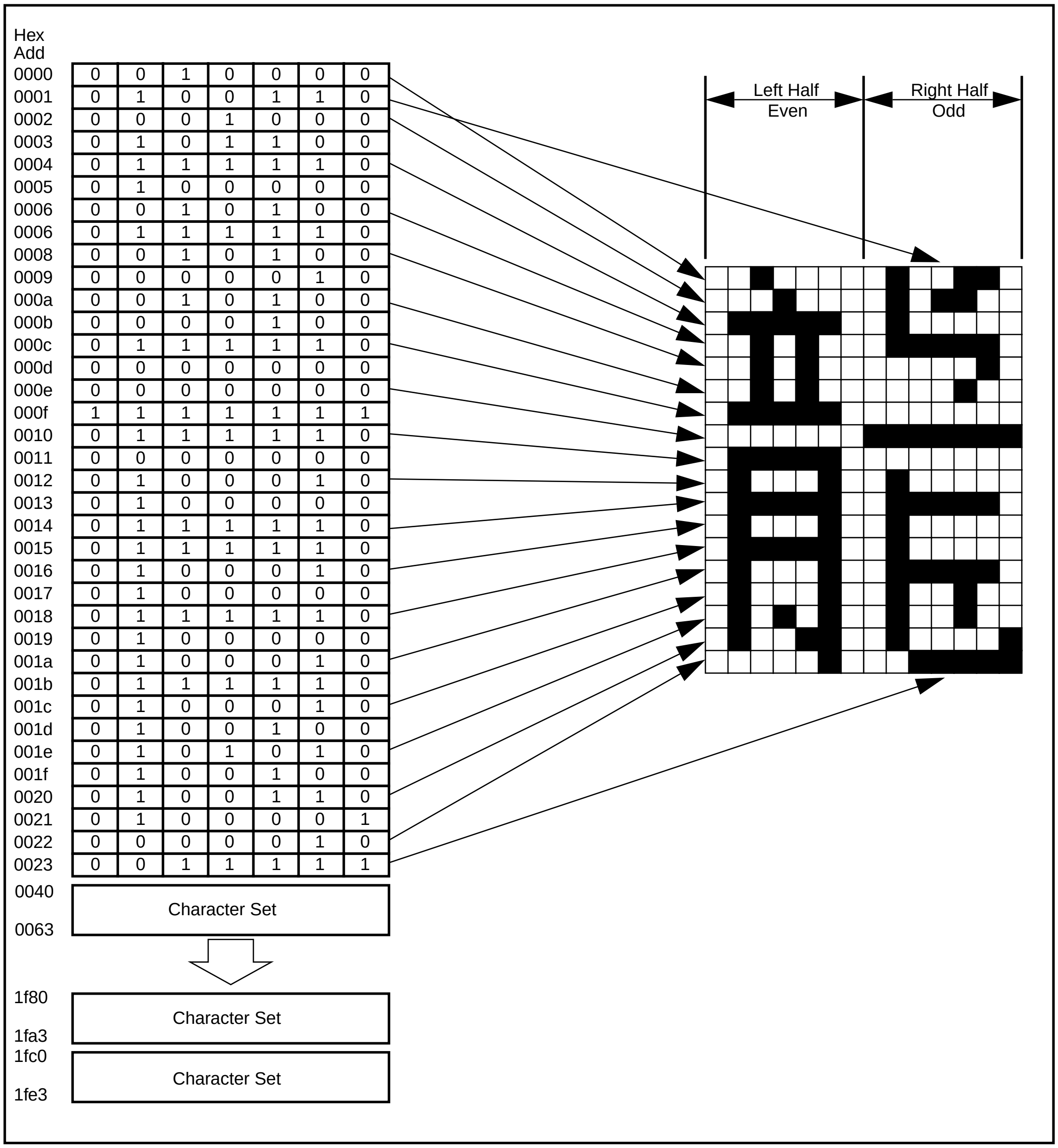


Figure 4-25. Character Pixel Map in CGROM (Example)

The character pixel map in Figure 4-25 represents one character. It is 14 pixels horizontal and 18 pixels vertical. Each row in the map is 7 bits long, half the width of the character scan line.

Even numbered rows of the map correspond to pixels on the left half of the character scan line; odd rows of the map correspond to pixels on the right half of the character scan line.

The Hex Add column is a hexadecimal number that serves as an address for the group of pixels from the starting point of the scan line. Addressing begins at 0000h and ends at 0023h.

Each bit in the map sets the foreground/background designation of the corresponding pixel: 0 - background, 1 - foreground pixel. The patterns formed by the bits comprise the characters that are displayed when the scan line is output to the screen.

Each of these character pixel maps is one character; 256 characters may be mapped. Each character starts with an offset of 40h from the previous character.

Multiple characters may be combined to form a large icon. Figure 4-26 shows an example. Each block marked by the darker grid lines is 14 x 18 pixels, one character.

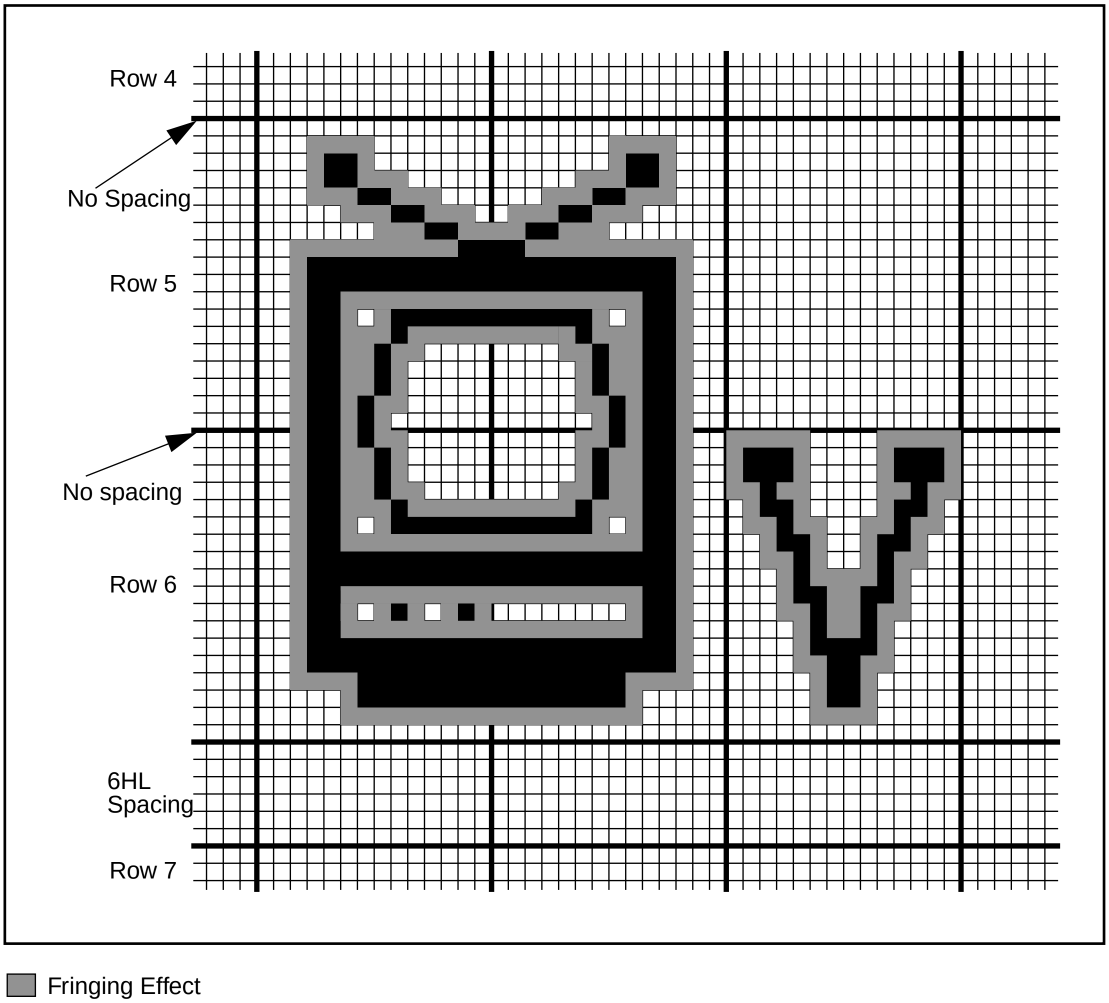


Figure 4-26. Icon Display

4.6.2 Character Size and Smoothing Effect

Z90230 supports two sizes of characters, 1X and 2X, as shown in Figure 4-26. The 2X size duplicates each pixel horizontally and vertically to reach the double size.

Smoothing is the enhancing of a character to improve its appearance. This effect can be applied only to 2X characters, and is enabled and disabled in DISP_ATTR: 03h: Bank A (4).

Check the effect of smoothing on 2X characters before finalizing OSD programming.

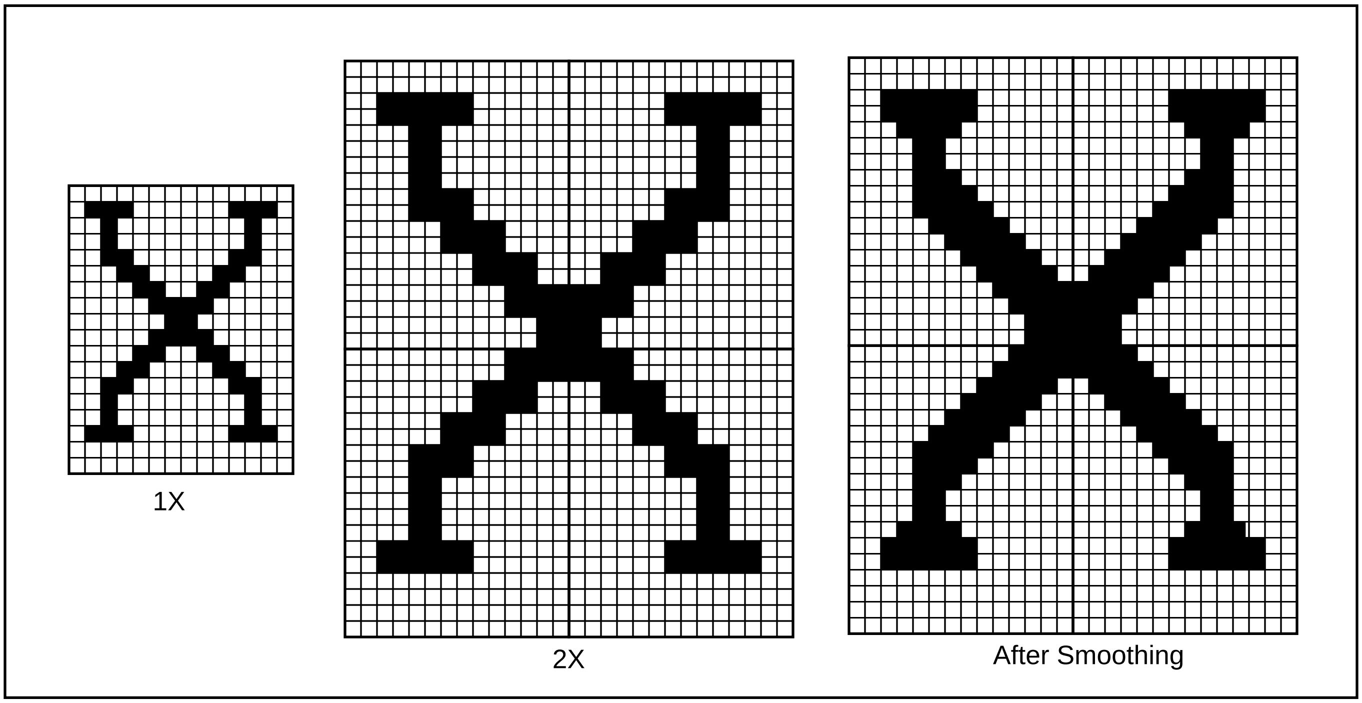


Figure 4-27. Smoothing

Figure 4-27 shows a character a 1X, 2X without smoothing, and 2X with smoothing.

4.6.3 Fringing Effect

Fringing is surrounding a character with color different from the foreground and background colors, as shown in Figure 4-26. Fringing adds visual appeal to the character presentation.

The fringing effect is enabled or disabled in DISP_ATTR: 03h: Bank A (5). The fringing color is set in INT_ST: 07h: Bank C (7) to either 0, the character background color, or to 1, a RGB color that is specified in INT_ST: 07h: Bank C (6,5,4). The eight RGB colors available for fringing and background are defined in Table 4-2.

Table 4-2. RGB Colors

R	G	B	Color
0	0	0	Black
0	0	1	Blue
0	1	0	Green
0	1	1	Cyan
1	0	0	Red
1	0	1	Magenta
1	1	0	Yellow
1	1	1	White

4.7 DISPLAY ATTRIBUTE CONTROL

Display attribute control determines characteristics of the screen display for the entire screen, not just the OSD area. The background that covers the entire screen is called the Master Background. Its color setting can be used to generate a blue screen when the TV signal is not present.

4.7.1 Display Attribute Register

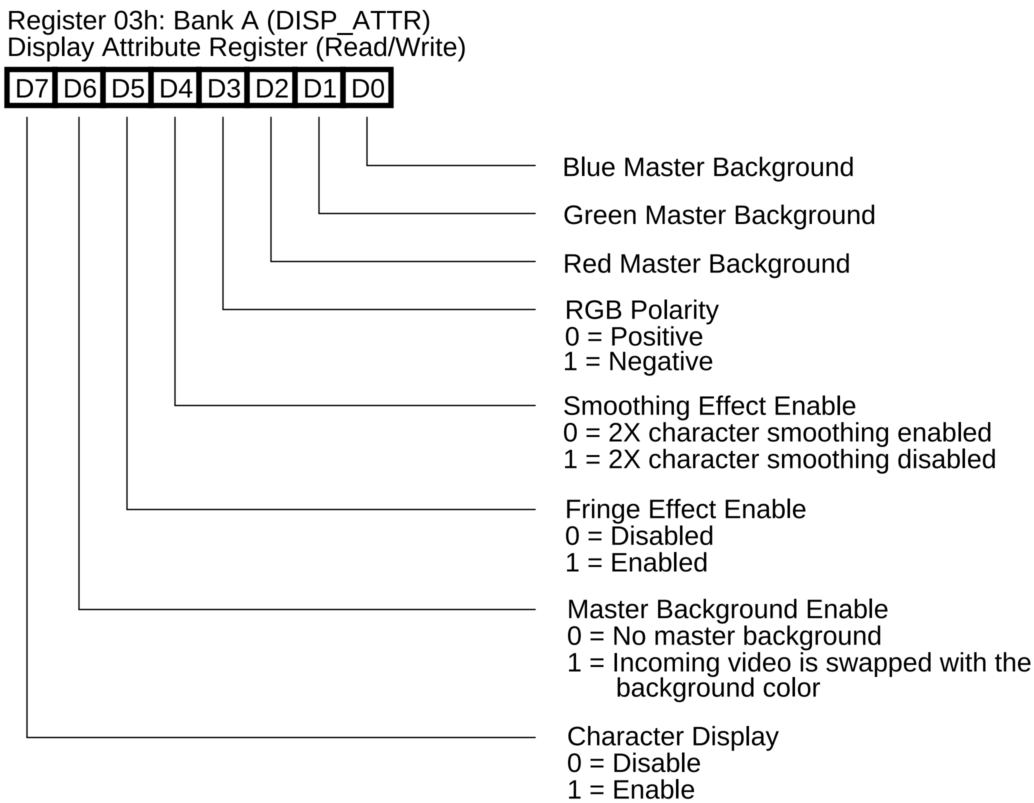


Figure 4-28. Display Attribute Register

Bit 7, Display Enable, disables or enables the use of foreground and background color, and therefore character display. When this bit is set to 0, effective space characters are sourced from the video RAM. Background On/Off and row background color are programmed independently. When bit 7 is set to 1, the actual video RAM characters are displayed.

Bit 6, Master Background Enable, disables or enables the use of a background color for the entire screen instead of the broadcast signal. If this bit is set to 1, the incoming video signal is blanked and the screen background is displayed in color according to the setting of the background color bits. The color is specified in bits 2, 1, 0. If bit 6 is set to 0, the incoming video signal is displayed.

Bit 5, Fringe Enable, sets the fringe effect On or Off.

Bit 4, Smoothing Effect Enable, sets smoothing On or Off, and is available only for 2X-size characters.

Bit 3, RGB Polarity, sets color polarity of OSD color output signals to positive or negative.

Bits 2, 1, and 0 form the color for the master background. The eight possible colors are the same as are listed in Table 4-2.

4.7.2 Video Refresh RAM Access

The Z90230 family of products supports 11-bit character data. Eight bits, D0 through D7, contain character data. Three additional bits, D8 through D10, contain color palette information.

Figure 4-29 contains the address map of VRAM for displaying 10 rows and 24 columns:

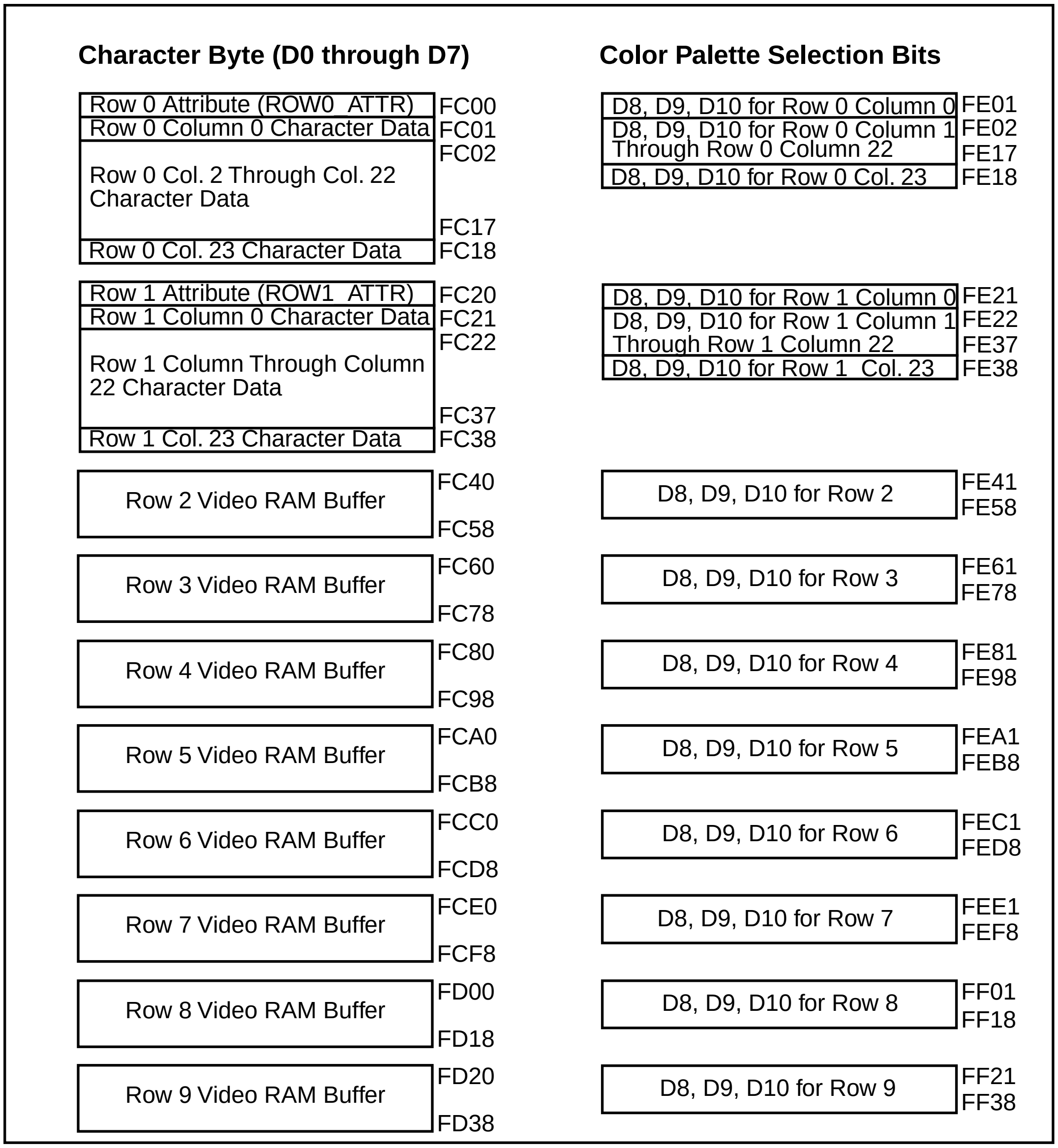


Figure 4-29. VRAM Address Map

Hardware processes the entire 11 bits of data at the same time it processes the OSD.

The Color Palette Selection Bits serve as a 3-bit Color Index to the color palette look-up table. Whenever software writes any Character Byte data (D0 - D7) into VRAM, it also takes the data in the color index register and writes the corresponding Color Palette Selection Bits (D8 - D10). These three bits can be updated separately (Figure 4-30).

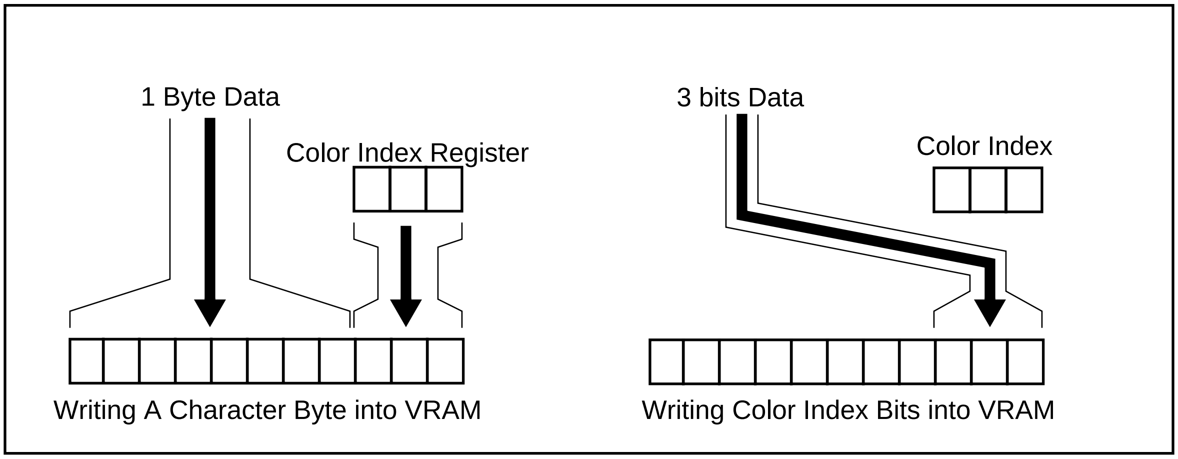


Figure 4-30. Color Palette Selection Bits Update

The Color Palette Selection Bits (D8 - D10) are decoded as follows:

Table 4-3. Color Palette Selection Bits

Color Index	Function
000	Selects background/foreground color in row attribute
001	Selects color palette 0 in color look-up table
010	Selects color palette 1 in color look-up table
011	Selects color palette 2 in color look-up table
100	Selects color palette 3 in color look-up table
101	Selects color palette 4 in color look-up table
110	Selects color palette 5 in color look-up table
111	Selects color palette 6 in color look-up table

There are eight different foreground/background palettes, including the 000 case that reads the color(s) from the ROW_ATTR register mapped into video RAM.

4.7.3 Color Table and Color Index Register

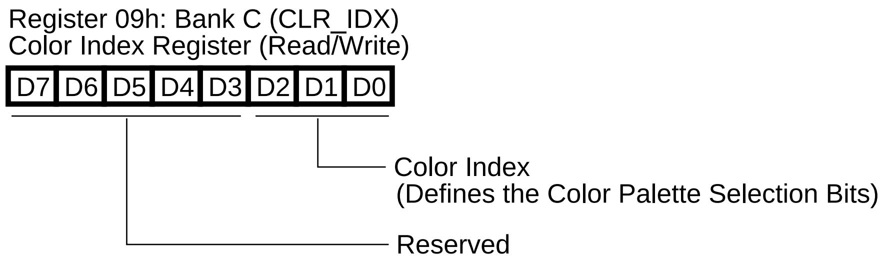


Figure 4-31. Color Index Register

When software reads the Color Index Register for the Color Index, the 5 unused bits (bits 7-3) return 1s.

When the Color Index has a value other than 000, the value indicates the number of the color palette that contains the RGB foreground and background colors to be displayed. In the Color Palette register descriptions below, the following notation is used:

Rnf	R - Red, n - Palette Number, f - Foreground
Rnb	R - Red, n - Palette Number, b - Background
Gnf	G - Green, n - Palette Number, f - Foreground
Gnb	G - Green, n - Palette Number, b - Background
Bnf	B - Blue, n - Palette Number, f - Foreground
Bnb	B - Blue, n - Palette Number, b - Background

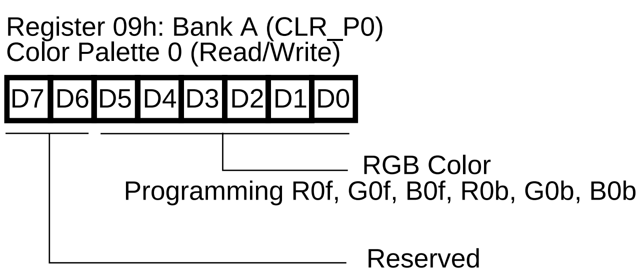


Figure 4-32. Color Palette 0

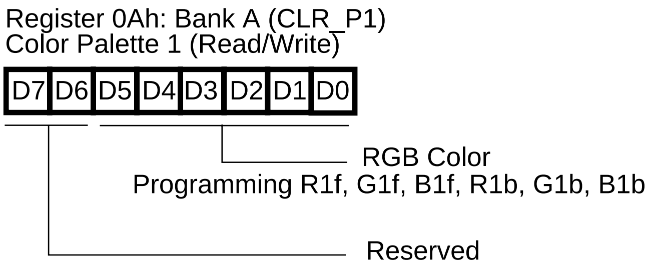


Figure 4-33. Color Palette 1

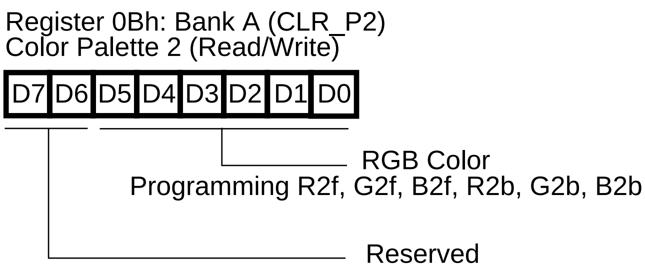


Figure 4-34. Color Palette 2

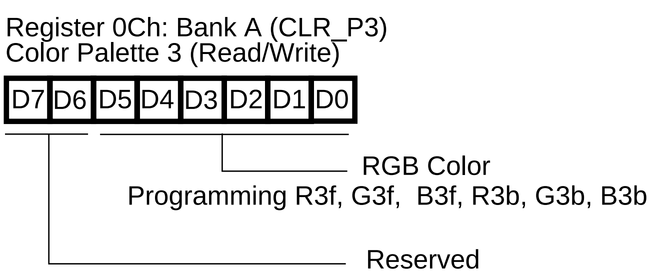


Figure 4-35. Color Palette 3

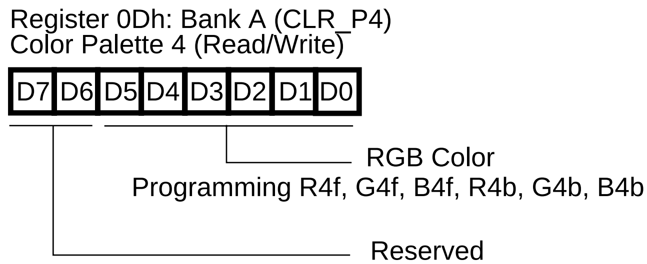


Figure 4-36. Color Palette 4

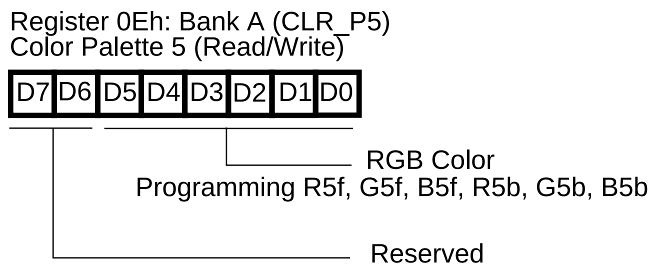


Figure 4-37. Color Palette 5

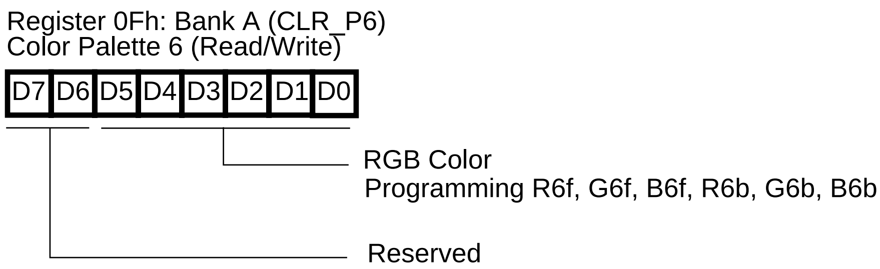


Figure 4-38. Color Palette 6

4.7.4 Row Attribute Register

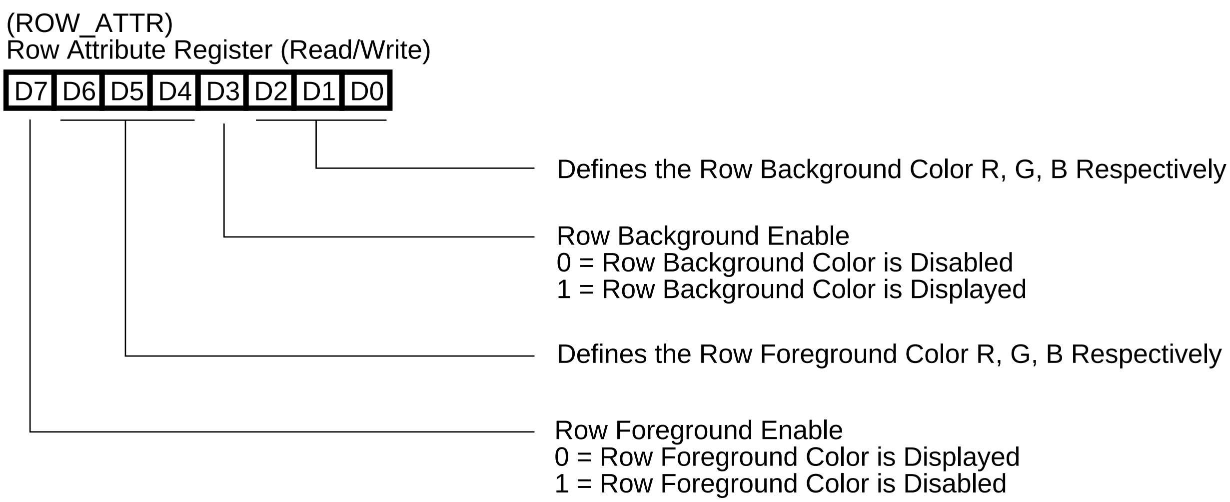


Figure 4-39. Row Attribute Register

The Row Attribute Register is mapped to VRAM, as shown in Figure 4-39. This register controls row background and foreground display. If the Color Index is set to 000, the display color is read from the Row Attribute Register.

Bit 7, Row Foreground Enable, enables or disables row foreground color.

Bits 6, 5, and 4, Row Foreground Color, designate the color of the characters displayed in the row.

Bit 3, Row Background Enable, disables or enables row background color.

Bits 2, 1, and 0, Row Background Color, designate the color of the row background.

4.8 HV INTERRUPT PROCESSING

An interrupt is issued at the beginning of a row and at the leading edge of the V_{SYNC} signal. The leading edge of the first H_{SYNC} of a row constitutes the beginning of a row. The Z90230 software tracks this cycle as two recurring events, the Horizontal (H_{SYNC}) Interrupt and the Vertical (V_{SYNC}) Interrupt.

A V_{SYNC} interrupt marks the time a new field of a frame is to be displayed, beginning with Row 0.

The display of subsequent rows coincides with the issuance of the H_{SYNC} interrupt. The interrupts mark the time when the display of a row or start of a field is to occur. Software must be ready to properly output the OSD when the interrupts are issued. Each text row is comprised of 18 scan lines. Each scan line takes 63.5 µs to be displayed. So, 1143 µs is the amount of time available for changing the programming for the next row. Double-size characters span 36 scan lines, allowing 2286 µs for programming the next row. Additional programming time is gained with inter-row spacing. During that time, VRAM is updated.

If the program has too much to display, black lines appear at the top of the screen.

The HV Interrupt Status Register keeps track of the type of interrupt that is issued—horizontal or vertical.

4.8.1 HV Interrupt Status Register

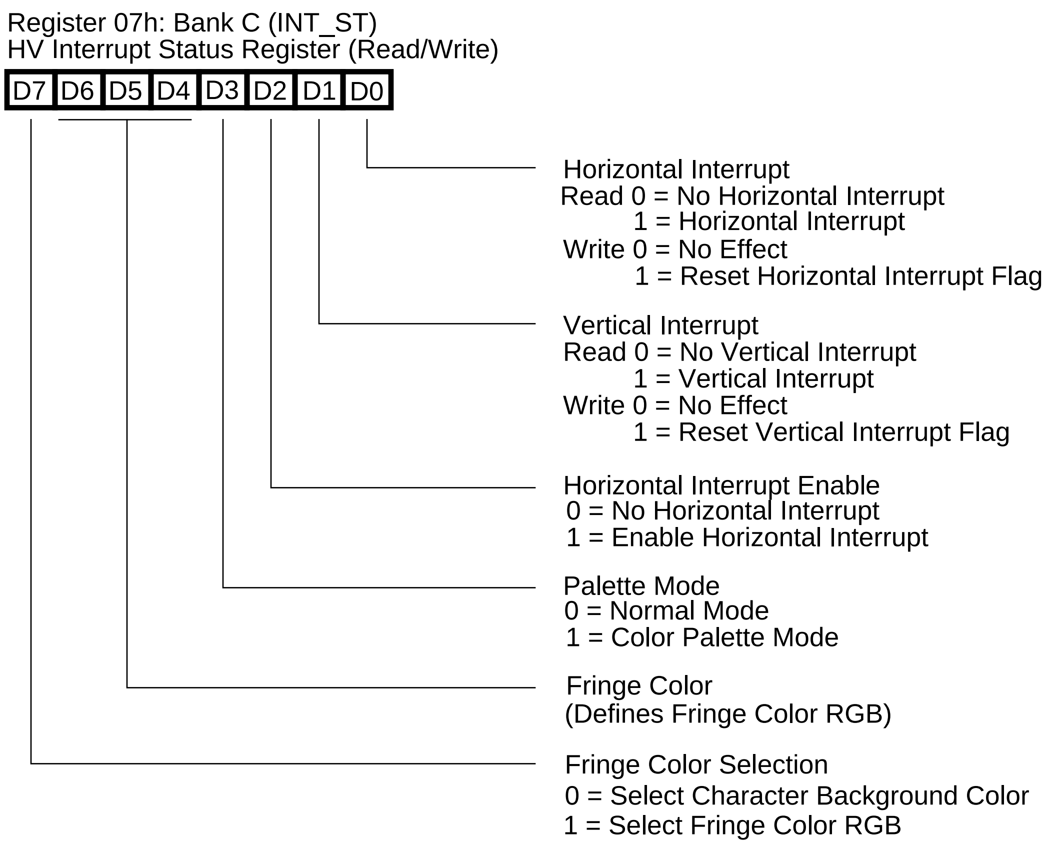


Figure 4-40. HV Interrupt Status Register

Bit 7, Fringe Color Selection, sets the fringe color to the background color or to a Red, Green, and Blue color that is specified in bits 6,5,4.

Bits 6, 5, and 4, Fringe Color, sets the Red, Green, and Blue values of the fringe color.

Bit 3, Palette Mode, sets color to Normal (8-bit) or VRAM (11-bit) Mode. When the value is 0 (Normal Mode), the color attribute of a row is controlled by values in the ROW_ATTR register, which is mapped in VRAM, but the Color Palette Selection Bits are ignored. When the Palette Mode value is 1, the Color Palette Selection Bits are used, unless they are set to 0s. In that case, the values in ROW_ATTR register are used.

Bit 2, Horizontal Interrupt Enable, disables or enables the horizontal (H_{SYNC}) interrupt.

Bit 1, Vertical Interrupt, has different meanings depending on its Read and Write status. In Read State, a value of 0 indicates that a vertical interrupt has not been issued; a value of 1 indicates that a vertical interrupt has been issued. In Write State, a value of 0 has no effect; a value of 1 resets the vertical interrupt flag.

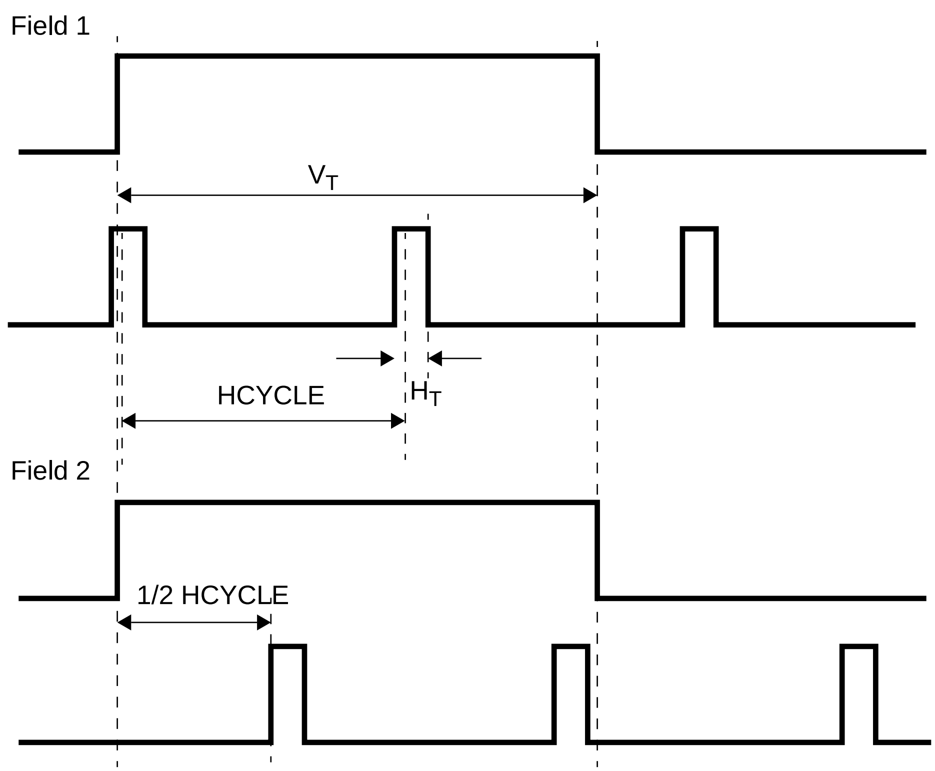
Bit 0, Horizontal Interrupt, has different meanings depending on its status. In Read State, a value of 0 indicates that a horizontal interrupt has not been issued; a value of 1 indicates that a horizontal interrupt has been issued. In Write State, a value of 0 has no effect; a value of 1 resets the horizontal interrupt flag.

When an interrupt is issued while another interrupt is being processed, the last-issued interrupt is pended. The interrupt-flag bit which is in service (the interrupt issued first) must be cleared or serviced before the pended interrupt can be processed (see SNDCLR(6)).

4.8.2 H_{SYNC} and V_{SYNC} Requirements

H_{SYNC} and V_{SYNC} must meet the all TV broadcasting specifications.

The minimum width of V_{SYNC} must conform to the following design:



As it is shown, V_T has to be larger than 1.5 x (HCYCLE + H_T). The same timing specification must be applied in negative polarity.

Figure 4-41. H_{SYNC} and V_{SYNC} Specification

4.9 DOT CLOCK OSCILLATOR

Dot clock oscillator for Z90230 family is generated by the LC network as shown in Figure 4-42.

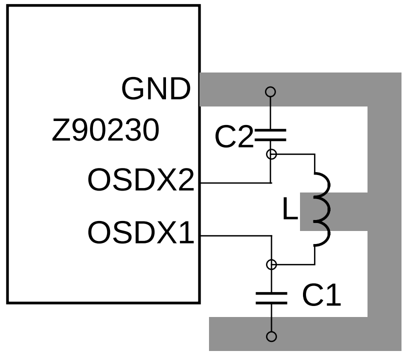


Figure 4-42. Dot Clock Oscillator

The frequency stays stable over V_{CC} and temperature. The oscillation frequency is determined by the equation:

$$Frequency = \frac{1}{2\pi\sqrt{LC_T}}$$

Figure 4-43. Oscillation Frequency

where L is the total inductance including parasitics and C_T is the total series capacitance including the parasitics. Simple series capacitance is calculated using the following equation.

$$\frac{1}{C_T} = \frac{1}{C_1} + \frac{1}{C_2}$$

Figure 4-44. Simple Series Capacitance

Care must be exercised in choosing LC values. Recommended values are L=27uH and C=22pF. (This value of C does not include routing capacitance.)

4.9.1 Layout

Traces connecting capacitors, inductor, and dot clock oscillator should be as short and wide as possible. This reduces parasitic inductance and resistance. The components (capacitors and inductor) should be placed close as possible to the dot-oscillator pins of the Z90230.

The traces from the oscillator pins of the IC and the ground side of the lead caps should be guarded from all other traces (clock, V_{CC}, address/data lines, and system ground) to reduce cross talk and noise injection.

CHAPTER 5
I^2C INTERFACE

5.1 I^2C-BUS CONCEPTS

Inter-Integrated Circuit (I^2C) is a serial interface. Two wires, serial data (SDATA) and serial clock (SCLK), carry information between the devices connected to the bus. Each device is recognized by a unique address and can operate as a transmitter and receiver, except as limited by the function of the device. A Master is a device which initiates a data transfer on the bus and generates the clock signals to enable the transfer. The non-initiating device is designated as the Slave.

The I^2C bus is a multi-Master bus. That is, more than one device capable of controlling the bus can be connected at any given time. Generation of clock signals on the I^2C bus is always the responsibility of Master devices. Each Master generates its own clock signals when transferring data. Bus clock signals from a Master can only be altered when they are stretched by a slow Slave device retaining the clock line at Low.

Both SDATA and SCLK are bidirectional lines, connected to a positive supply voltage via pull-up resistors. When the bus is free, both lines are High. The output stages of devices connected to the bus must have an open-drain or open collector in order to perform the wired AND function.

5.2 DATA VALIDITY

Data on the SDATA line must be stable during the High clock period (Figure 5-1). The High and Low state of the data line can only change when the clock signal on the SCLK line is Low.

5.3 START AND STOP CONDITIONS

Within the procedure of the I^2C bus, unique situations arise which are defined as Start and Stop conditions. One such unique case is when a High to Low transition of the SDATA line while the SCLK line is High. This situation indicates a Start condition. A Low to High transition of the SDATA line while SCLK line is High defines a Stop condition. Start and Stop conditions are always generated by the Master. The bus is considered to be busy after the Start condition. The bus is free again after the Stop condition.

5.4 DATA TRANSFER

Data transfer follows the procedure illustrated in Figure 5-1. At the Start condition, the address of a Slave device is sent. This address is 7 bits long followed by an eighth bit which is a data direction bit ($R/\overline{W}$)—a 0 indicates a transmission (Write), a 1 indicates a request for data (Read).

A data transfer is always terminated by a Stop condition generated by the Master. However, if a Master still wishes to communicate on the bus, it can generate a repeated Start condition to another Slave address without generating a Stop condition. This type of data transfer is called *combined format*. Some examples of combined format include:

- A Master transmits data to a Slave and then reads data from the same Slave.
- A Master transmits data to one Slave and then transmits data to another Slave.
- 10-bit and 7-bit addressing can be combined in one serial transfer.

For some types of serial memory, a combined format is the only way to read data from a precise location.

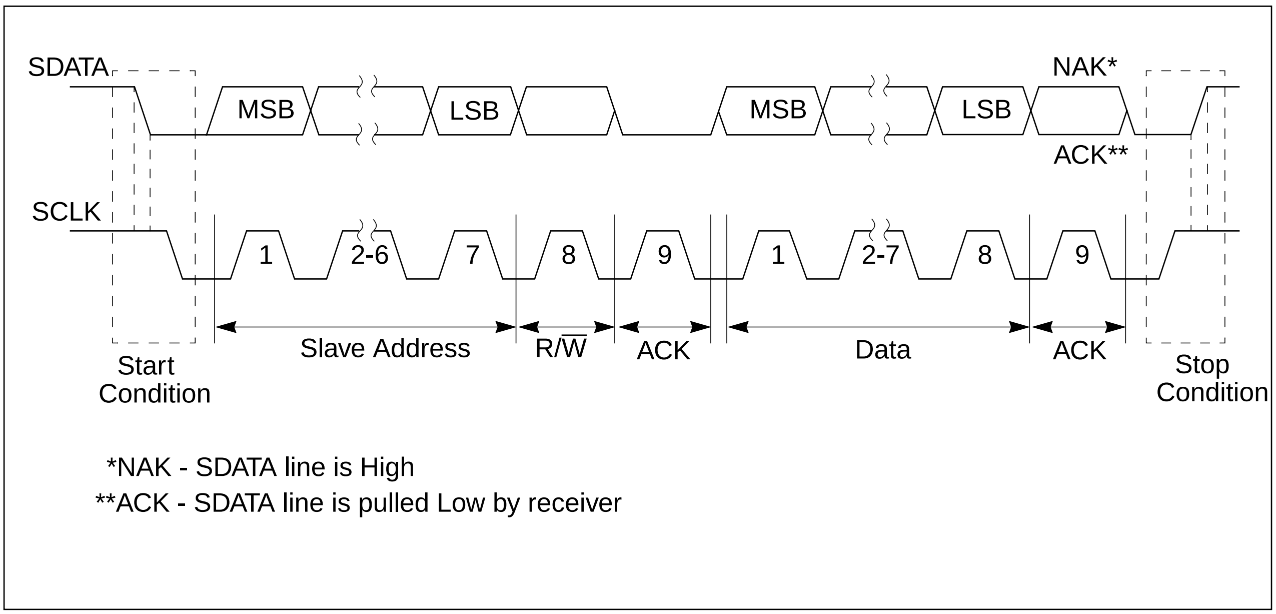


Figure 5-1. Data Transfer

5.5 BYTE FORMAT

The number of bytes transmitted or received by a Master during one communication session is unrestricted. Each byte must be followed by an acknowledgment bit. Data is transferred with the most significant bit (MSB) first. If the Slave is not capable to receive or transmit another complete byte of data in one continuous stream, it can hold the SCLK line Low to force the Master into a wait state. Data transfer is automatically resumed when the Slave releases the SCLK line. The Slave may start to hold clock line Low only during the lower period of the clock pulse generated by the Master.

5.6 ACKNOWLEDGE

Acknowledgment of a data transfer is obligatory. The acknowledge-related clock pulse is generated by the Master. The transmitter releases the SDATA line (changing it to High) prior to the acknowledge clock pulse. The receiver changes the SDATA line during the acknowledge clock pulse (ACK) so it remains stable and Low during the upper period of the clock pulse (Figure 5-1).

When a Slave-receiver does not acknowledge (NAK) a transmitted byte, the data line is left High by the Slave during the acknowledge clock pulse, and the Master can generate a Stop condition to abort the transfer.

The Master/receiver must signal the end of data to the Slave/transmitter by not generating an acknowledge on the last byte that was transferred from the Slave. The Slave/transmitter must release the data line to allow the Master to generate the Stop condition.

5.7 Z90230 FAMILY I²C MASTER INTERFACE

Z90230 family has the hardware module which supports the I²C Master interface. Bus arbitration and Masters' arbitration logic is not implemented; in other words, the Z90230 family is designed for a *single Master* application.

The I²C interface can be configured to run at 4 different transfer speeds defined by bits (1,0) in the I²C Control Register (I²C_CNTL: 0Ch, Bank: C).

In order to suppress possible glitches on both DATA and SCLK lines, digital filters with time constant equal to $3T_{sclk}$ is implemented on all inputs of the I²C bus interface. The Z90230 family has two separate I²C busses which share the same control and data registers.

The I²C module is enabled by setting bit (2) in the I²C_CNTL register to 1. This bit blocks out I²C logic if it is set to 0 (Figure 5-2). To prevent switching the I²C bus during activation, bits (7,6) of the Port 2 Data Register for I²C selection 1 (bits (5,4) of Port 2 Data Register for I²C selection 0) should be set to 1 before the I²C module is enabled.

Notes:

1. When the I²C module is enabled, pins used as I²C must be configured as output in the Port 2 Mode Register (P2M: F6h). If P27/P26 or P25/P24 are used as I²C pins, then these pins are automatically set to open-drain mode.
2. Port 2 must be configured in standard drive mode (PCON: 00h: Bank F) when the I²C interface is active.

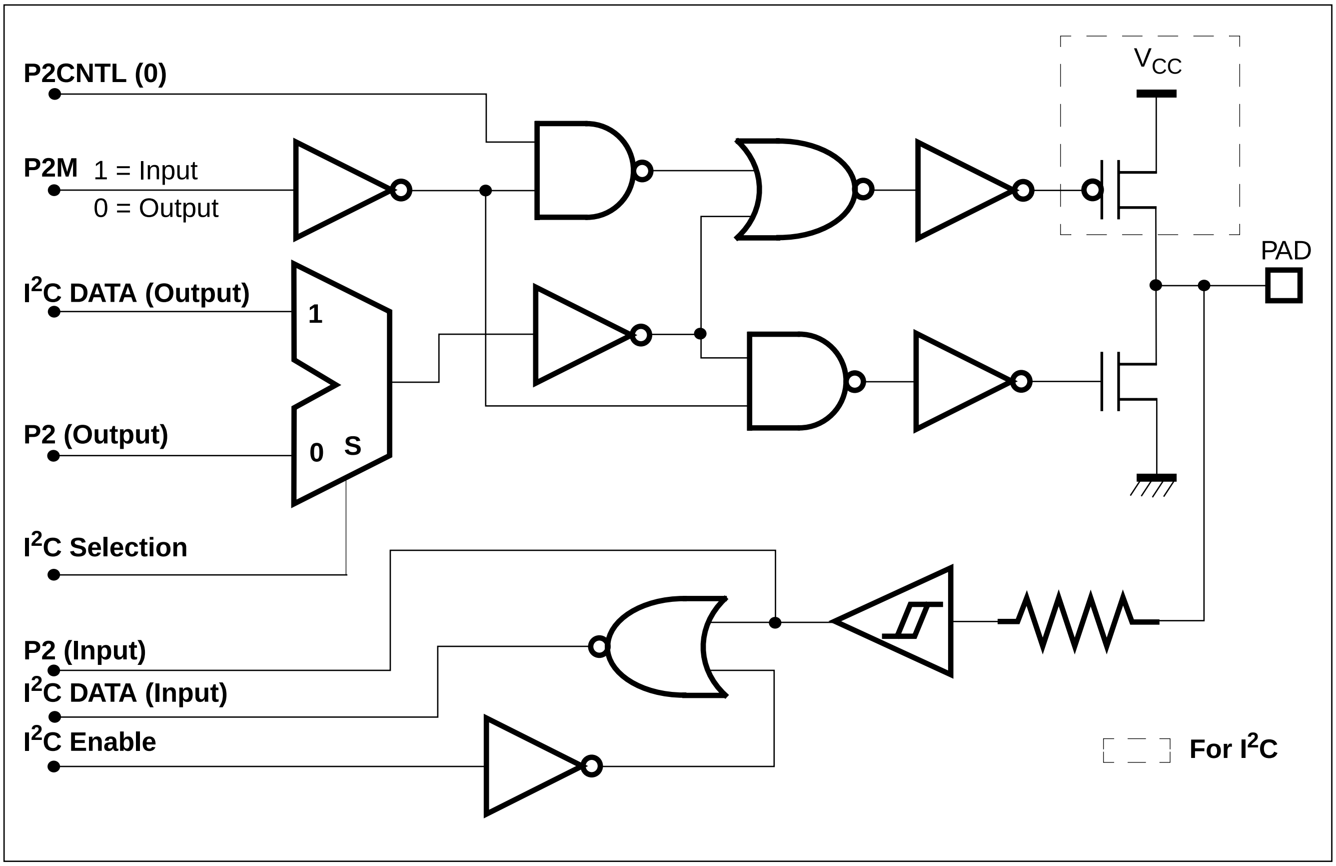


Figure 5-2. Bidirectional Port Pin Pad Multiplexed with I²C Port

5.7.1 Master I^2C Control Register

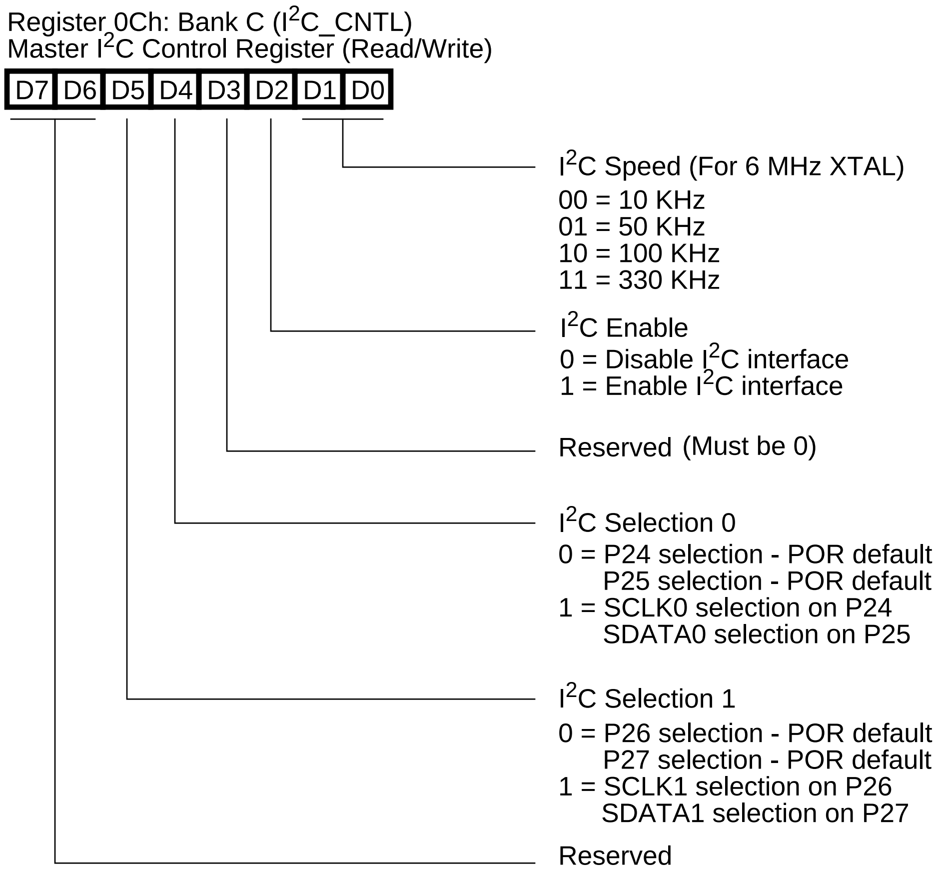


Figure 5-3. Master I^2C Control Register

If bits D4 and D5 both equal 1, then the I^2C Selection 0 prevails.

5.8 SOFTWARE CONTROL OF THE I^{2}C INTERFACE

Software controls the I^{2}C module by writing appropriate commands into the I^{2}C Command Register (I^{2}C_CMD: 0Bh: 0Ch).

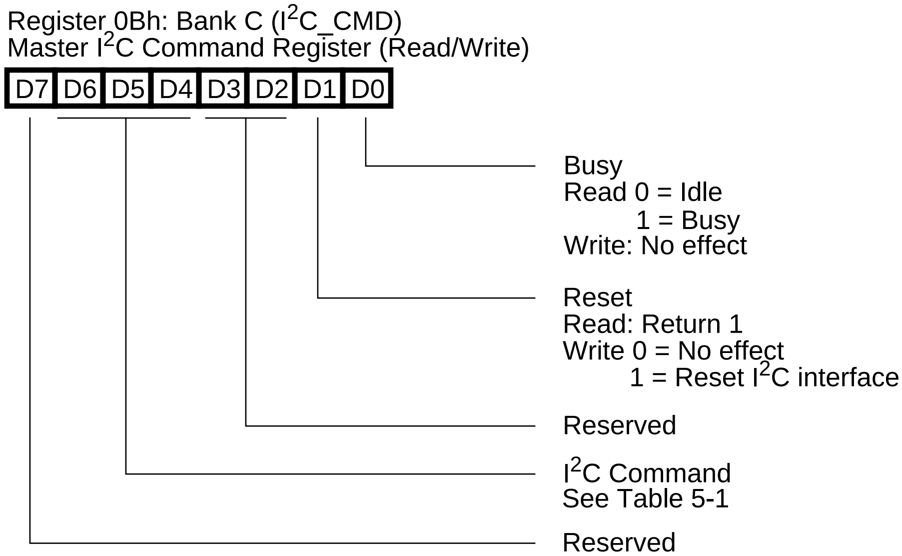


Figure 5-4. Master I^{2}C Command Register

The commands in Table 5-1 are the values that go into D6, D5, and D4 of the Master I^{2}C Command Register.

Software puts data for transmission into I^{2}C Data Register (I^{2}C_DATA: 0Ah: 0Ch) and reads received data from it. Bit 7 of this register is used as an acknowledgment bit during receiving data from a Slave. Bit 0 of I^{2}C_DATA register contains an acknowledgment bit generated by Slave.

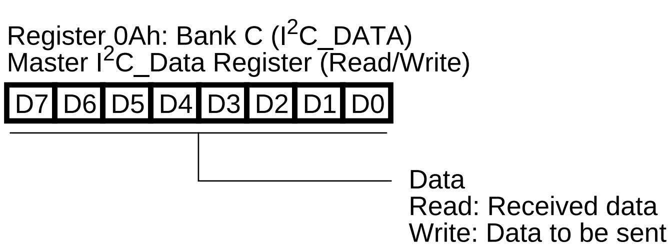


Figure 5-5. Master I^{2}C Data Register

In order to have appropriate sequence of I^{2}C command executed by the I^{2}C module software has to check Busy Bit (bit[0] in the I2C_CMD). The busy bit is set to 1 at the beginning of each command executed by the I^{2}C module, and stays 1 for the entire command cycle. Then, it changes to 0.

Flowcharts of writing and reading a data frame for I^{2}C devices with 7-bit addresses are shown in Figure 5-6 and Figure 5-7.

The same algorithms can be used for I^{2}C devices with 10-bit addresses. The 10-bit addressing does not affect the existing 7-bit addressing. A special combination (11110xx) for the first 7 bits of the first byte following a START bit is reserved for 10-bit addressing only. The special combination can not be used as an address of a device with 7-bit addressing. The last two bits (xx) of this combination are the two most-significant bits (MSBs) of the 10-bit address. The eighth bit of the first byte is a data direction bit (R/W). It has same meaning as in 7-

bit addressing—a 0 indicates a transmission (Write), a 1 indicates request for data (Read). The second byte contains remaining 8 bits of the 10-bit address. Then Master sends or receives data as in 7-bit addressing mode.

Table 5-1. Master I^{2}C Bus Interface Commands

Command	Description
000	Send a Start bit followed by the address byte specified in the I^{2}C data register, then fetch the acknowledgment bit in I^{2}C_DATA (0). Used to initialize communication. Nine SCLK cycles are generated.
001	Send the byte of data specified in the I^{2}C data register, then fetch an acknowledgment bit stored in bit 0. Used in a Write frame. Nine SCLK cycles are generated.
010	Send bit 7 of I^{2}C_DATA register as an acknowledgment bit (ACK: (0XXXXXXX), NAK: (1XXXXXXX)), then receive a data byte. Used in a Read frame when the next data byte is expected. Nine SCLK cycles are generated. Received data is read in the I^{2}C data register.
011	Send bit 7 of I^{2}C_DATA register as an acknowledgment bit (ACK: (0XXXXXXX), NAK: (1XXXXXXX). Used in a Read frame. One SCLK cycle is generated.
10X	Null operation. Must be used with a Reset bit.
110	Received one data byte. Used in a Read frame in order to receive the first data byte after an address byte is transmitted. Eight SCLK cycles are generated.
111	Send Stop bit. One SCLK cycles are generated.

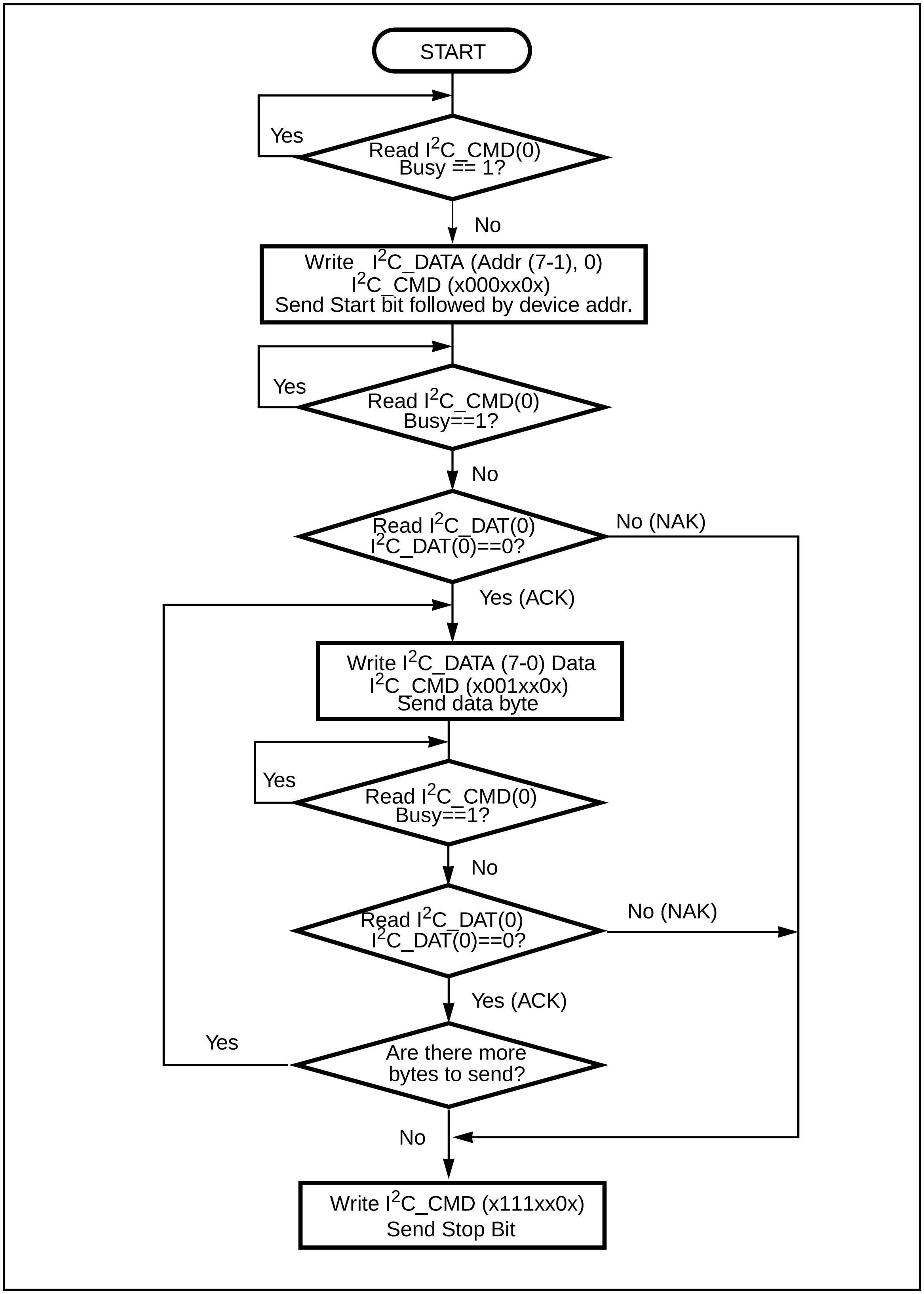


Figure 5-6. Data Frame Write Flowchart

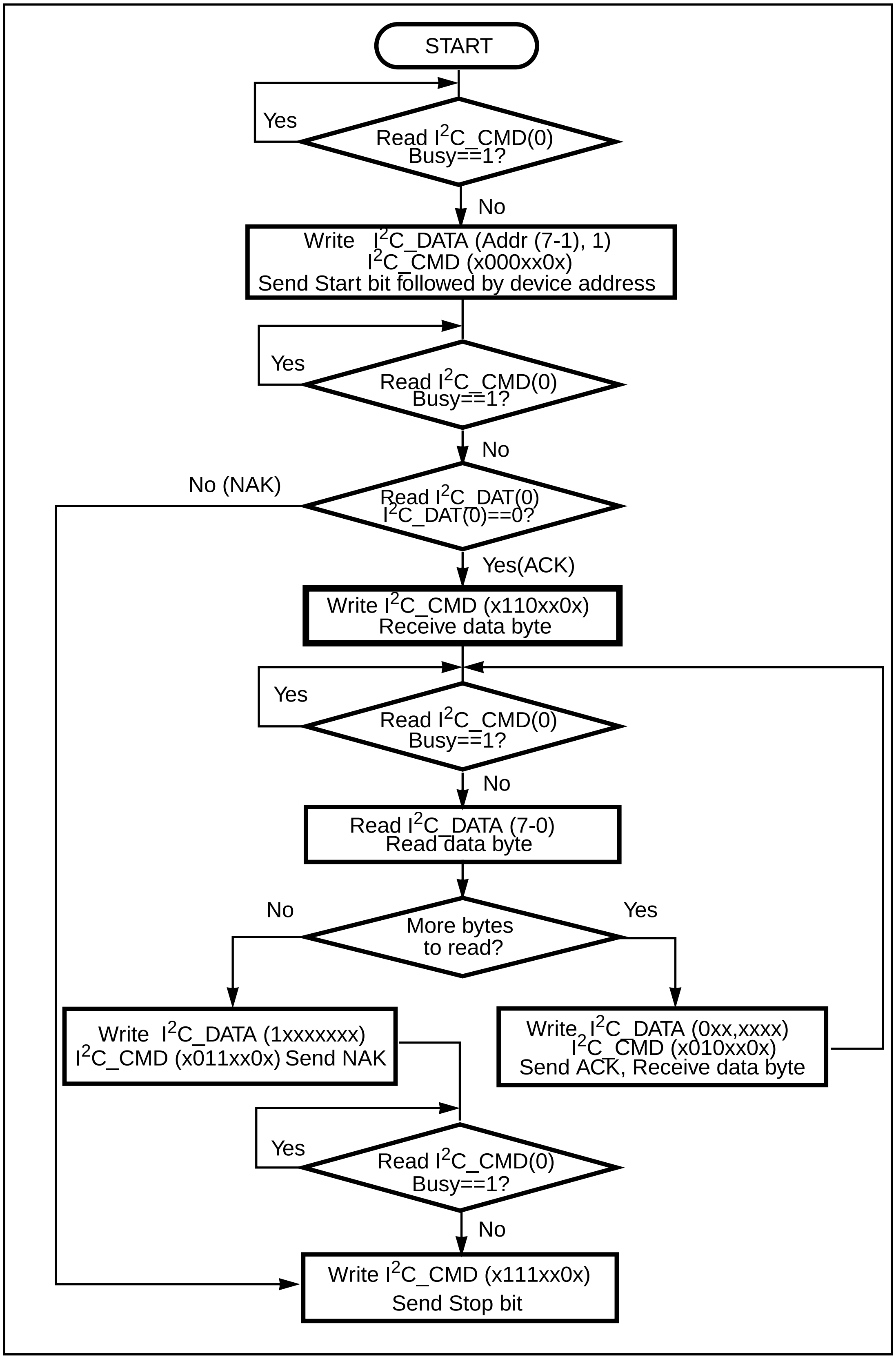


Figure 5-7. Data Frame Read Flowchart

CHAPTER 6
INPUT/OUTPUT PORTS

6.1 INPUT/OUTPUT PORTS

There are 20 input/output (I/O) ports. In addition, seven pulse-width modulators (PWM), PWM 1 through PWM 6, and PWM 11, can be configured as regular output ports. The maximum number of I/O ports available is 27. Please refer to the port bank and number carefully for exact addressing and access.

6.1.1 Port Configuration Register

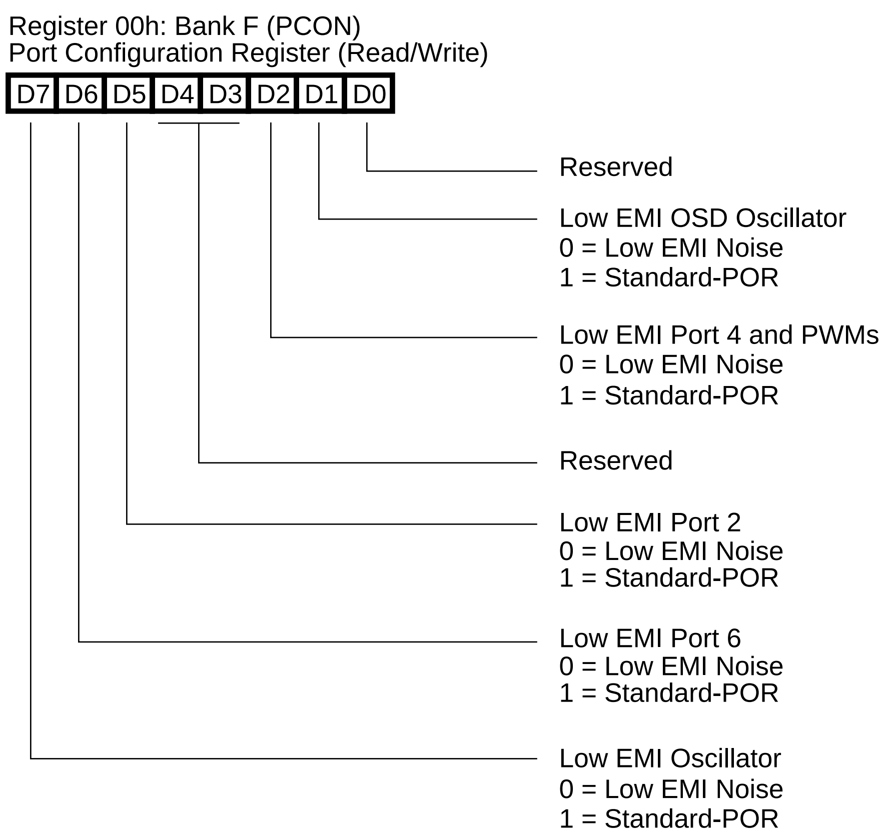


Figure 6-1. Port Configuration Register

Ports 2, 4, and 6 may be set for Standard or Low EMI. The Low EMI option can also be selected for the microcontroller oscillator or OSD oscillator. Standard (1) is the High setting. Following Power-On Reset, Bits 2, 5, 6, 7 each has a value of 1.

6.1.2 Port 2 Mode Register

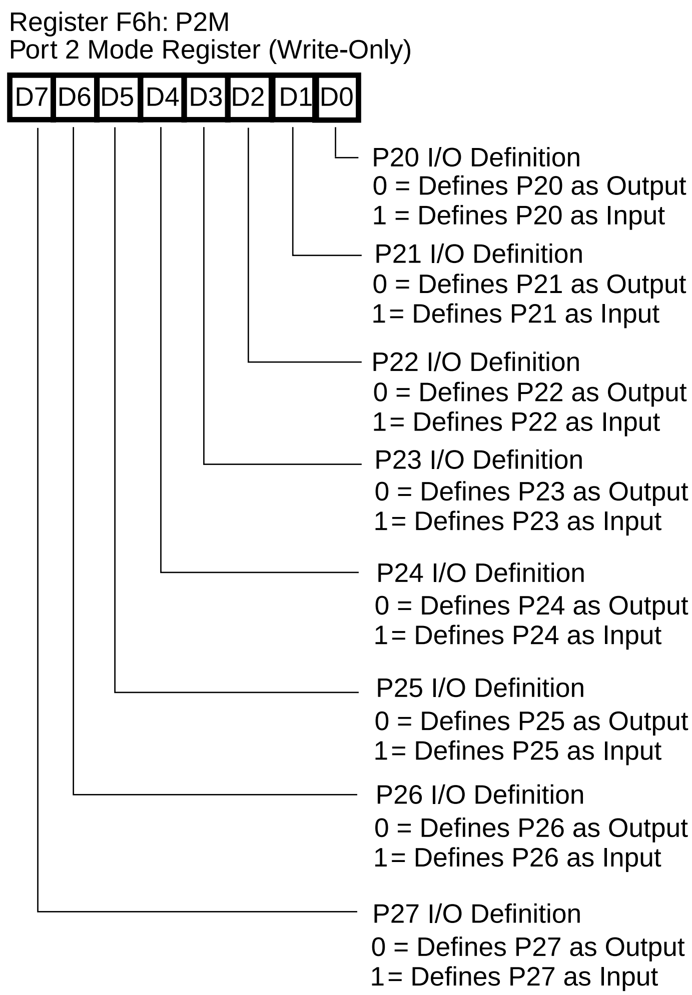


Figure 6-2. Port 2 Mode Register

When P27/P26 or P25/P24 are used as I^2C pins, then these pins are automatically set to open-drain mode.

6.1.3 Port 2 Data Register

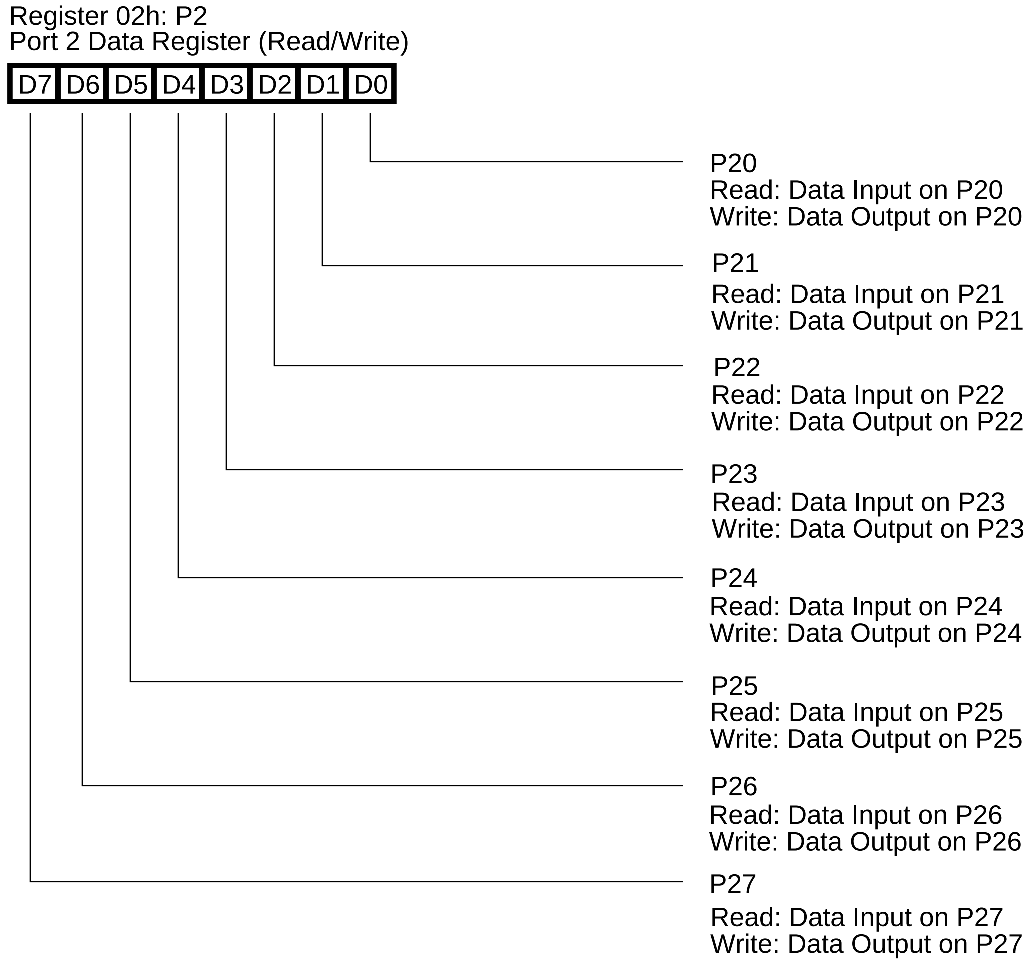


Figure 6-3. Port 2 Data Register

6.1.4 Port 4 Pin-Out Selection Register

Bits 5, 4, 3, and 2 control the configuration of multiplexed pins 20, 19, 18, and 17. If a bit is reset to 0, the pin functions as a PWM output port. This value is the default following a Power-On Reset. If a bit is set to 1, the pin functions as a programmable regular input/output port.

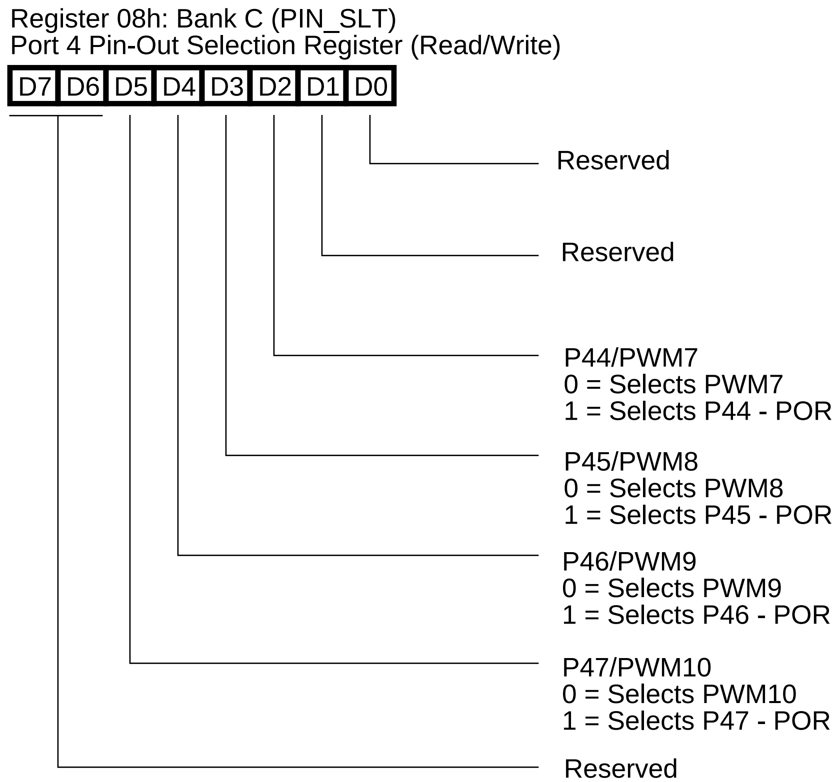


Figure 6-4. Port 4 Pin-Out Selection Register

6.1.5 Port 4 Data Register

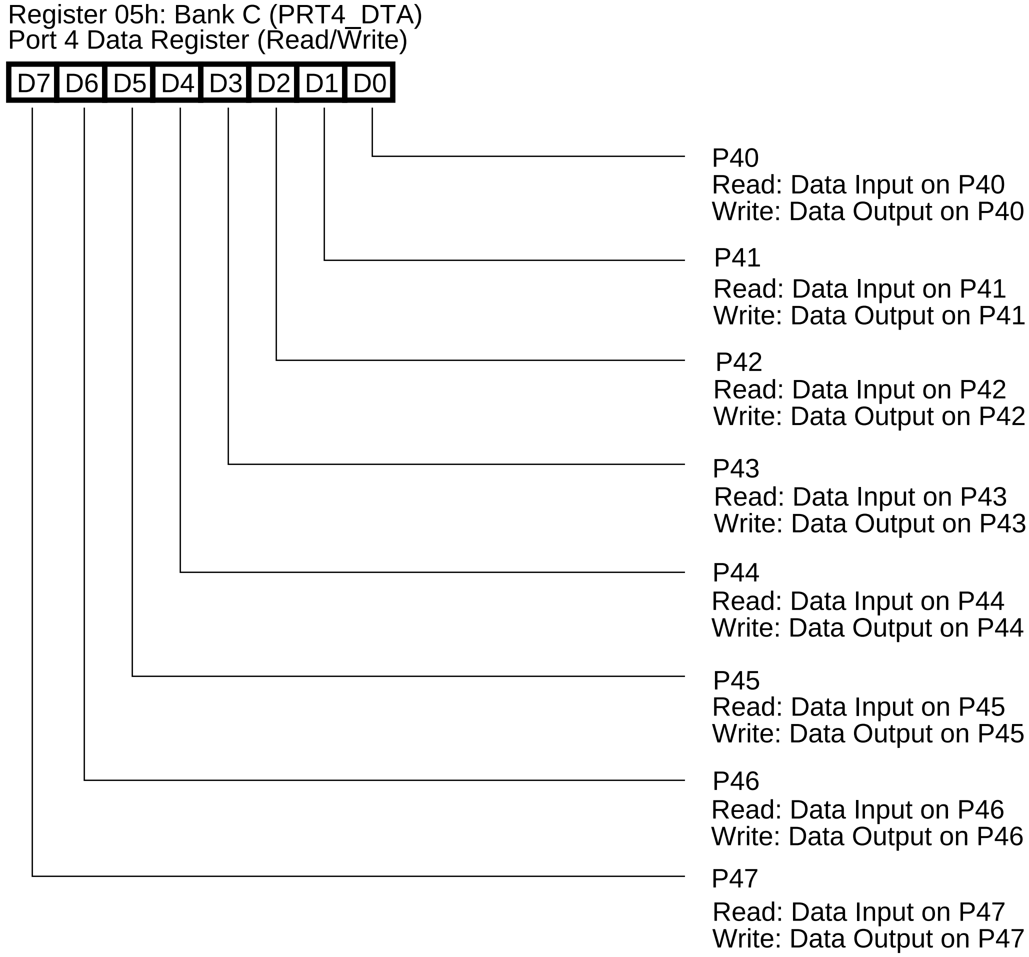


Figure 6-5. Port 4 Data Register

6.1.6 Port 4 Direction Control Register

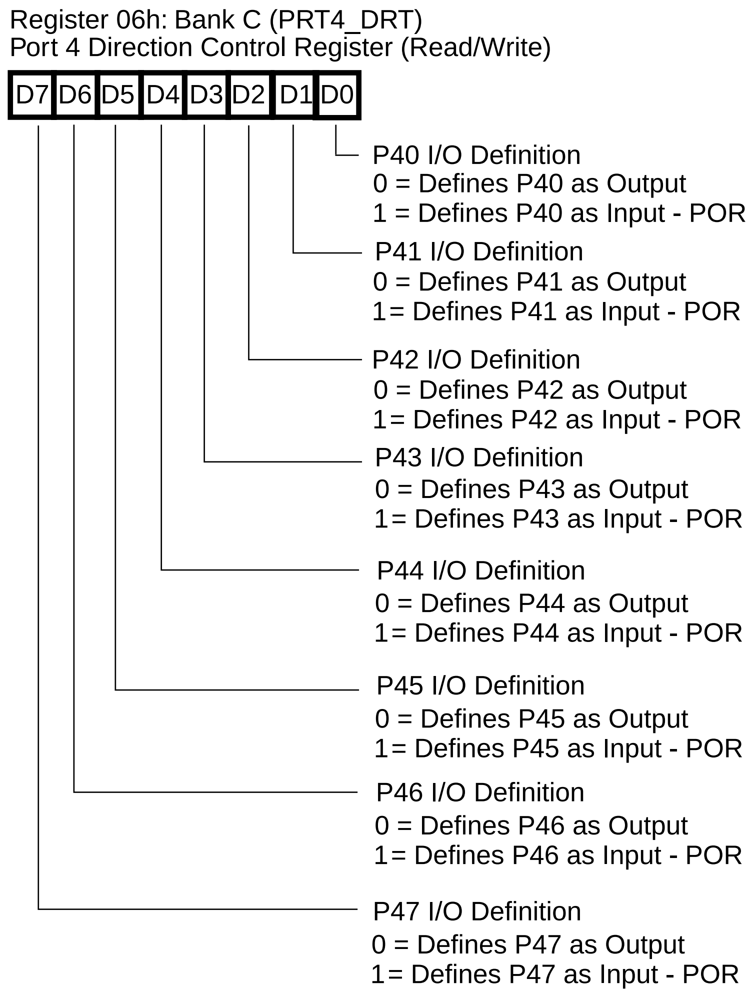


Figure 6-6. Port 4 Direction Control Register

6.1.7 Port 5 - PWM Mode Register

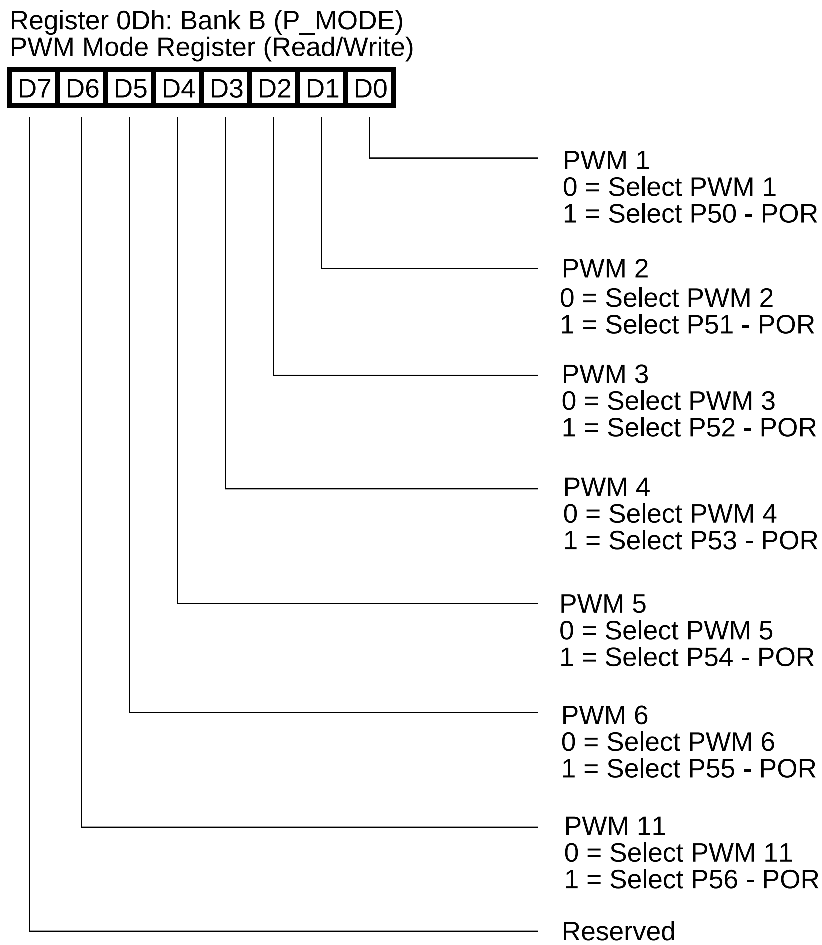


Figure 6-7. PWM Mode Register

6.1.8 Port 5 Data Register

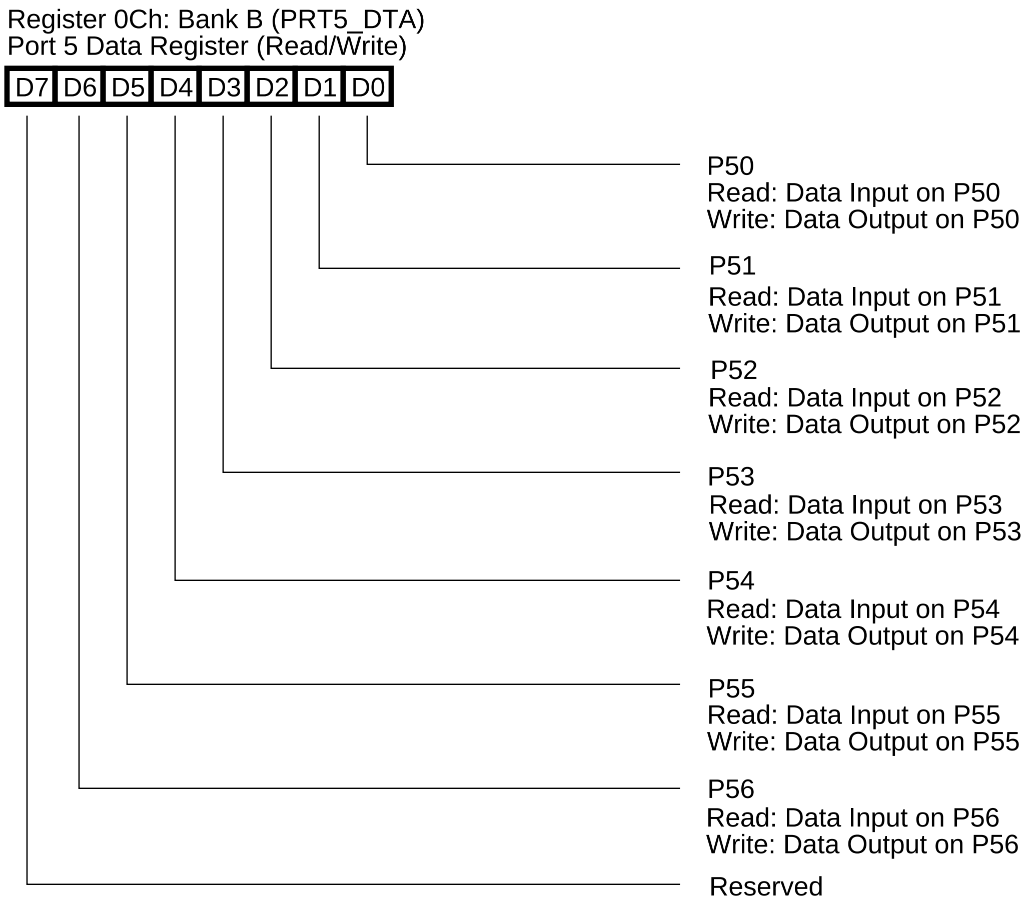


Figure 6-8. Port 5 Data Register

6.1.9 Port 5 Direction Control Register

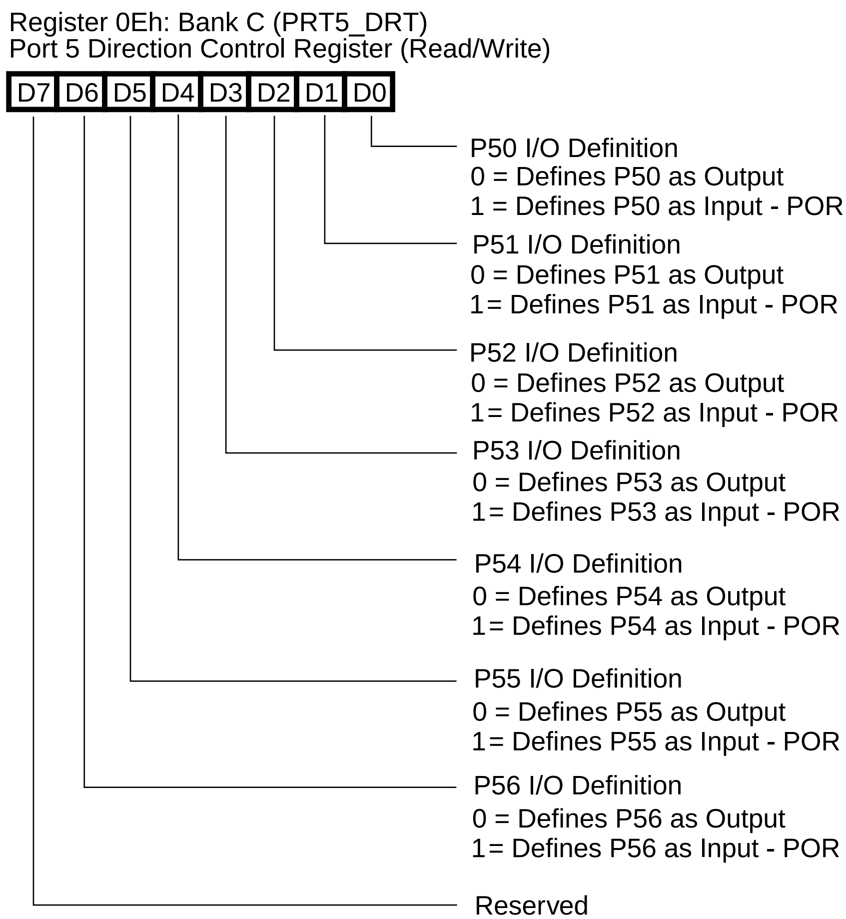


Figure 6-9. Port 5 Direction Control Register

The Port 5 Direction Control Register identifies each bit as output (0) or input (1) data.

6.1.10 Port 6 Data Register

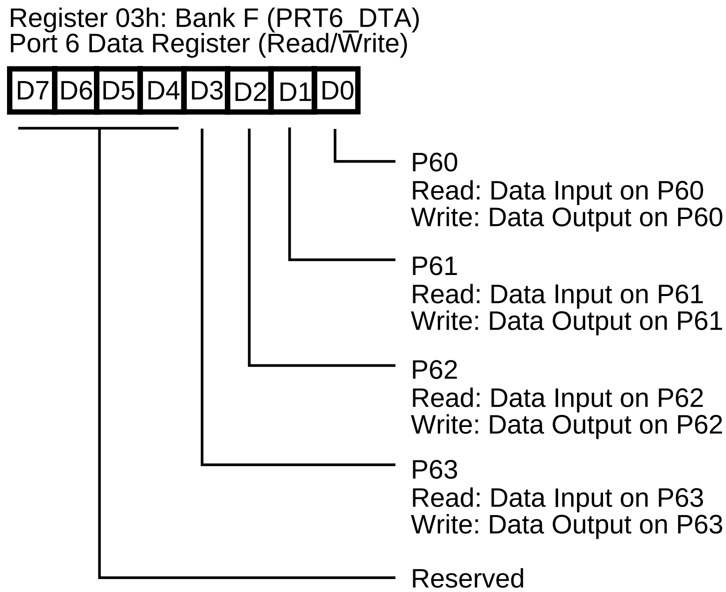


Figure 6-10. Port 6 Data Register

6.1.11 Port 6 Direction Control Register

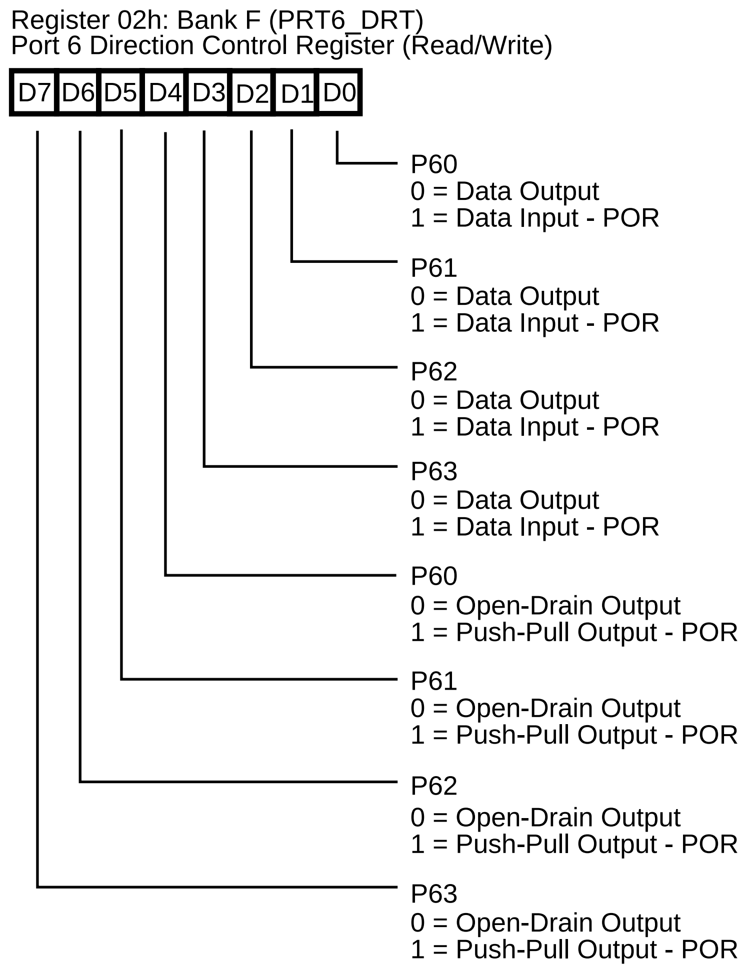


Figure 6-11. Port 6 Direction Control Register

CHAPTER 7
INFRARED INTERFACE

7.1 INFRARED INTERFACE

The Z90230 family easily supports the Infrared (IR) Remote Control interface with a minimum of software overhead.

The Digital Television Controller (DTC) has a hardware IR capture module which consists of :

- Timer Control Register0 (TCR0: 01h: Bank C)
- Timer Control Register1 (TCR1: 02h: Bank C)
- IR Capture Register0 (IR_CP0: 03h: Bank C)
- IR Capture Register1(IR_CP1: 04h: Bank C)

The IR capture registers are the Low and High bytes of the IR Capture Counter.

After an IR interrupt occurs, the software clears the corresponding interrupt flag bit.

Two bytes of data are received through the Infrared (IR) Interface. The upper byte, bits 15-8, is stored in IR Capture Register 1. The lower byte, bits 7-0, is stored in IR Capture Register 0.

When an IR interrupt occurs, the IR capture registers contain the amount of time passed from the previous IR interrupt if bit 0 in the TCR0 is set to 0. If bit 0 is set to 1, the IR capture registers contain the amount of time passed from the last overflow of the IR capture counter. The IR interrupt flags are reset by the IR interrupt service routine software.

7.1.1 Timer Control Register 0

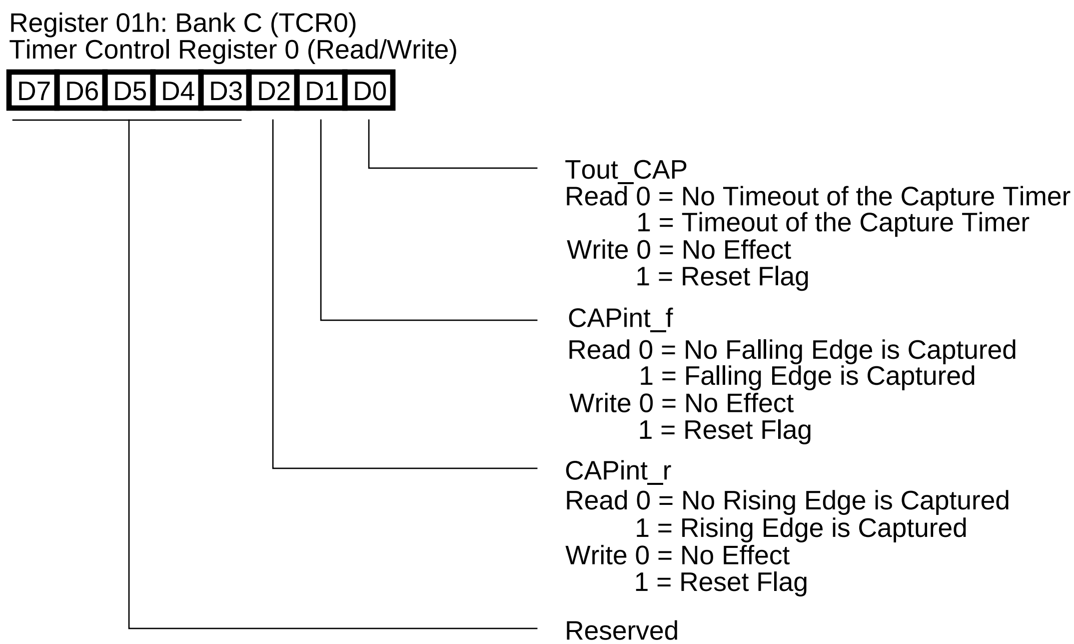


Figure 7-1. Timer Control Register 0

Rising edge (falling edge) interrupt is preserved even when a falling edge (rising edge) interrupt occurs. But it is overridden by a second rising edge (falling edge) if the second one occurs before the first rising edge (falling edge) is serviced. Preservation of the interrupt means that it will generate the hardware interrupt after the first interrupt is serviced when two different (rising edge/falling edge) interrupts are already ON.

During the interrupt service routine, software must read the contents of this register. Then it checks which bit is set to 1, indicating the type of edge which generated the interrupt (see Table 7-1).

Table 7-1. IR Interrupt Captured Values

D2D1D0	Edge	Timeout
100	Rising	No
101	Rising	Yes
010	Falling	No
011	Falling	Yes
110	Rising/Falling	No
111	Rising/Falling	Yes

7.1.2 Timer Control Register 1

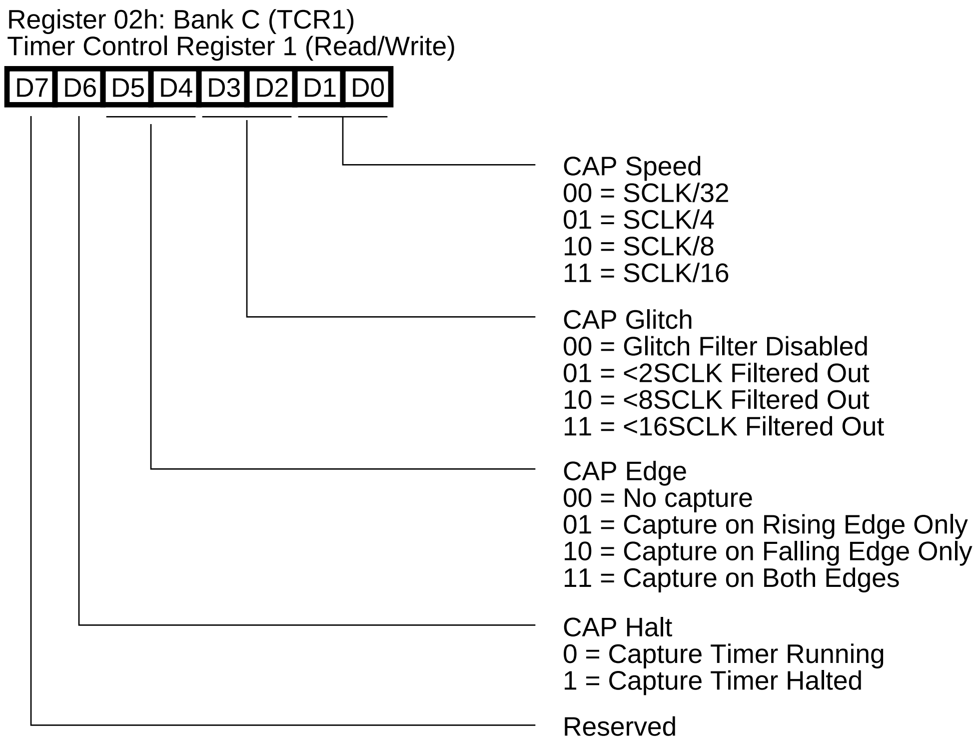


Figure 7-2. Timer Control Register 1

Bit 7 is Reserved.

Bit 6 resets the IR Capture Timer. To stop the timer, set this bit to 1. To start the timer, set the bit to 0.

Bits 5 and 4 set the IR Capture Edge. The rising edge, the falling edge, or both edges of an input signal can be used as the source of IR interrupts. If both edges are set as interrupt sources, Timer Control Register 0 (TCR0: 01h: bank C) must be read and checked by the Interrupt Service Routine (ISR) in order to identify which edge has been captured.

Bits 3 and 2 contain a time constant used in a digital filter to process the IR Capture module in order to prevent errors.

Bits 1 and 0 set the IR Capture Counter to one of four different speeds:

Table 7-2. IR Capture Timer Speed Setting

TCR1 (1, 0)	Timer Speed
00	SCLK/32
01	SCLK/4
10	SCLK/8
11	SCLK/16

The IR capture counter is driven by the clock generated by dividing the system clock of the Z90230.

7.1.3 IR Capture Register 0

Figure 7-3. IR Capture Register

7.1.4 IR Capture Register 1

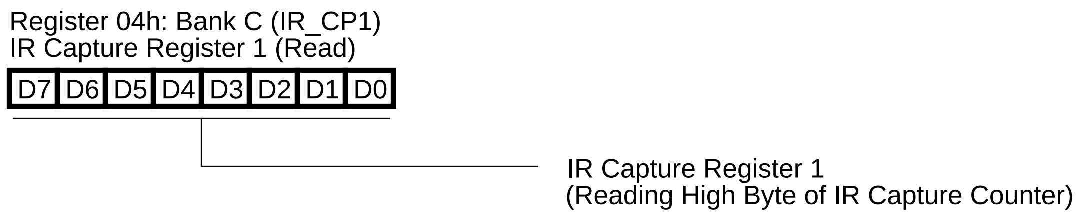


Figure 7-4. IR Capture Register 1

7.1.5 IR Decoding

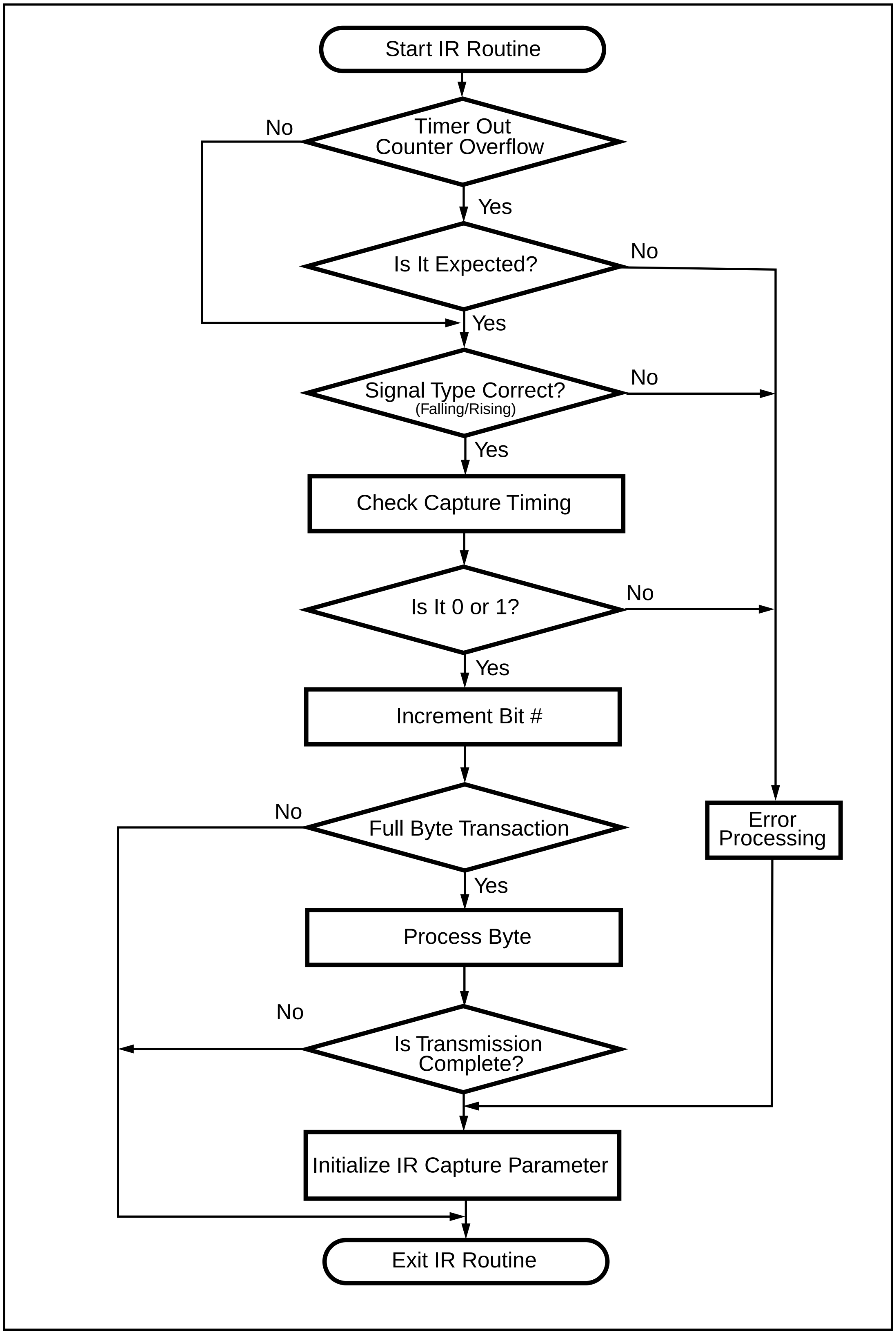


Figure 7-5. IR Decoding Flowchart Example

Note: This flow chart does not include processing a start bit, which some protocols require.

The Full Byte Transaction conditional statement does not necessarily require a full byte. It is the user's responsibility to determine the number of bits required to decode the IR signal.

CHAPTER 8
PULSE WIDTH MODULATORS

8.1 PULSE WIDTH MODULATORS

The Z90230 family has 11 Pulse Width Modulator channels. PWMs 1 through 10 have 6-bit resolution and are typically used for audio and video level control. PWM 11 has 14-bit resolution and is typically used for voltage synthesis tuning.

The PWM control registers are mapped into ERF Bank B:

Table 8-1. Expanded Register File Bank B

Register	Register Function	Working Register
E	PRT5_DRT	R14
D	P_MODE	R13
C	PRT5_DTA	R12
B	PWM10	R11
A	PWM9	R10
9	PWM8	R9
8	PWM7	R8
7	PWM6	R7
6	PWM5	R6
5	PWM4	R5
4	PWM3	R4
3	PWM2	R3
2	PWM1	R2
1	PWM11 Low Byte	R1
0	PWM11 High Byte	R0

PWM 11 uses two registers to accommodate its 14-bit resolution.

There are 6-bit and 14-bit binary counters for the 6-bit and 14-bit PWMs. The counter value is compared with the respective PWM register value and an output flip-flop is set to 1 when the values match. The flip-flop is reset to 0 when the counter section reaches zero. All PWM registers and their respective output flip-flops are cleared to zero after reset; therefore, all PWM ports are set to Low as an initial state.

Table 8-2. Pulse Width Modulator Pin Functional Description Example

8.1.1 PWM Mode Register

PWM Mode Register controls the setting of the multiplexed pins 1-7. These pins can be configured to function as PWM output ports or regular output ports. If a bit is reset to 0, the pin outputs the PWM signal. This setting is the default value following a Power-On Reset. If a bit is set to 1, the pin serves as a regular output port.

Bit 7 is reserved.

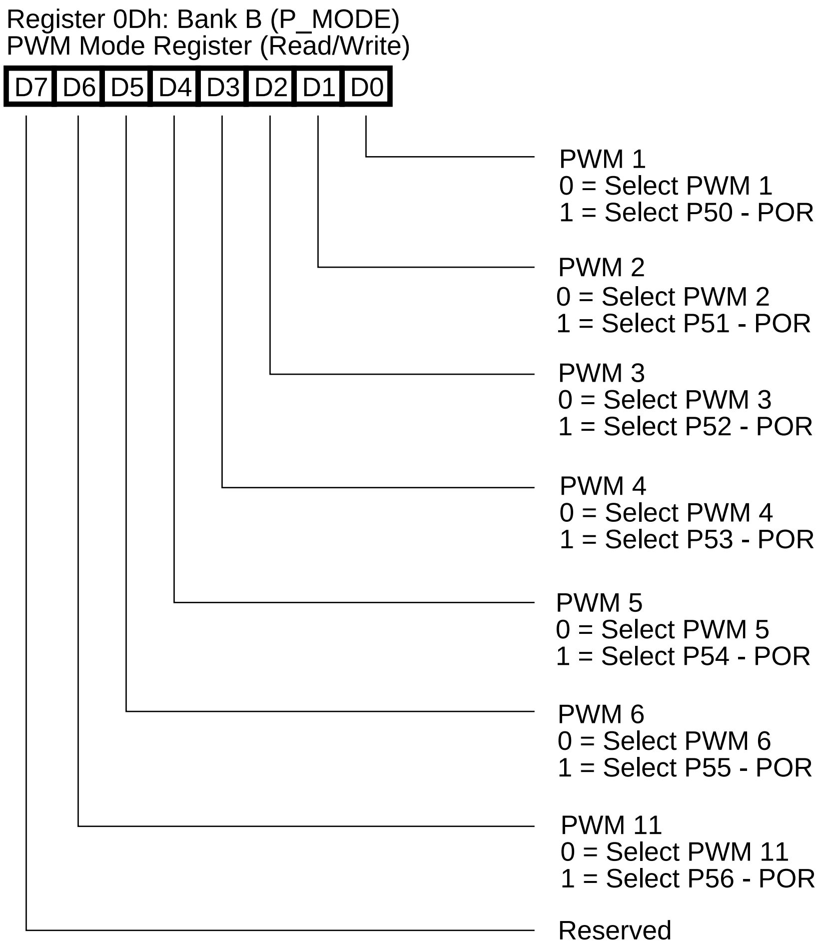


Figure 8-1. PWM Mode Register

8.1.2 Port 4 Pin-Out Selection Register

Bits 5, 4, 3, and 2 control the configuration of multiplexed pins 20, 19, 18, and 17. If a bit is reset to 0, the pin functions as a PWM output port. This value is the default following a Power-On Reset. If a bit is set to 1, the pin functions as a programmable regular input/output port.

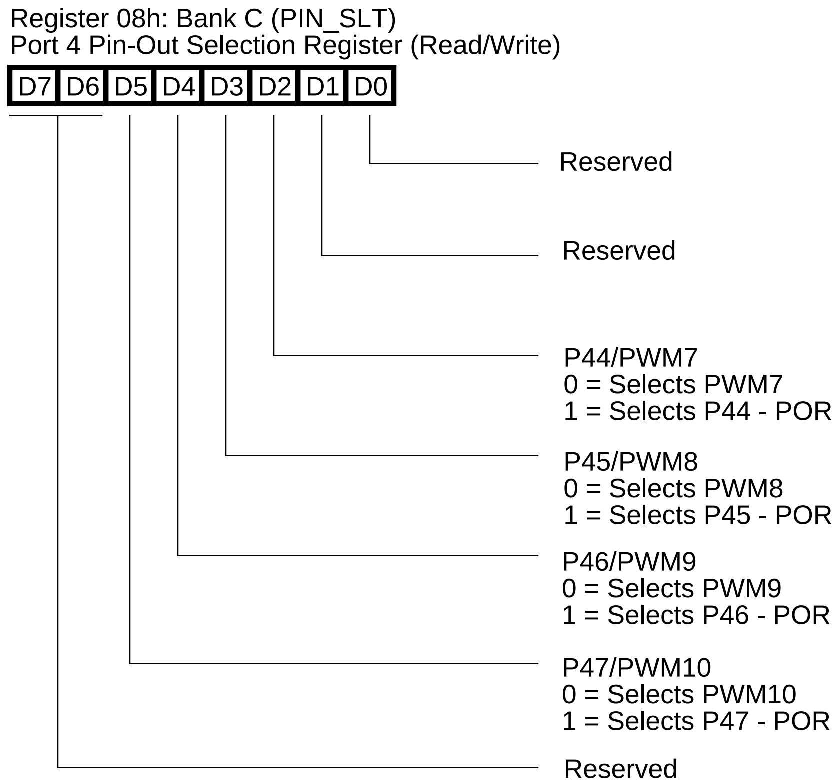


Figure 8-2. Port 4 Pin-Out Selection Register

8.1.3 PWM1 through PWM11

Two data registers (PWM11_H and PWM11_L) hold the 14-bit PWM11 ratio. The upper 7 bits controls the width of the distributed pulse. The lower 7 bits distribute the minimum resolution pulse in the various time slots. By using this technique, the pseudo-repetition of frequency is raised up to 128 times faster than ordinary pulse width modulation.

There are 128 time slots which start from time slot 7Fh to 0 because a 14-bit binary down counter is used. When the glitch exceeds 127 pulses, the upper 7 bits take precedence and fill 128 pulses of the same width in different locations. Generating the pulse-train output requires the following equation: Time slot (Fts) and one cycle of frequency (F14).

Fdp (Distribution pulse frequency)=XTAL/128 (Hz)

Fts (Time slot frequency) = XTAL/128 (Hz)

F14 (a cycle/frequency) = XTAL /16384 (Hz)

When the 6-bit data is 00h, the PWM output is Low. The maximum value is 3Fh and emits High DC-level output. A selected PWM cycle/frequency is shown in the following equation:

F6 (a cycle/frequency) = XTAL/16/64 (Hz)

Figures show various timing pulses and resultant frequencies for the 6- and 14-bit PWMs.

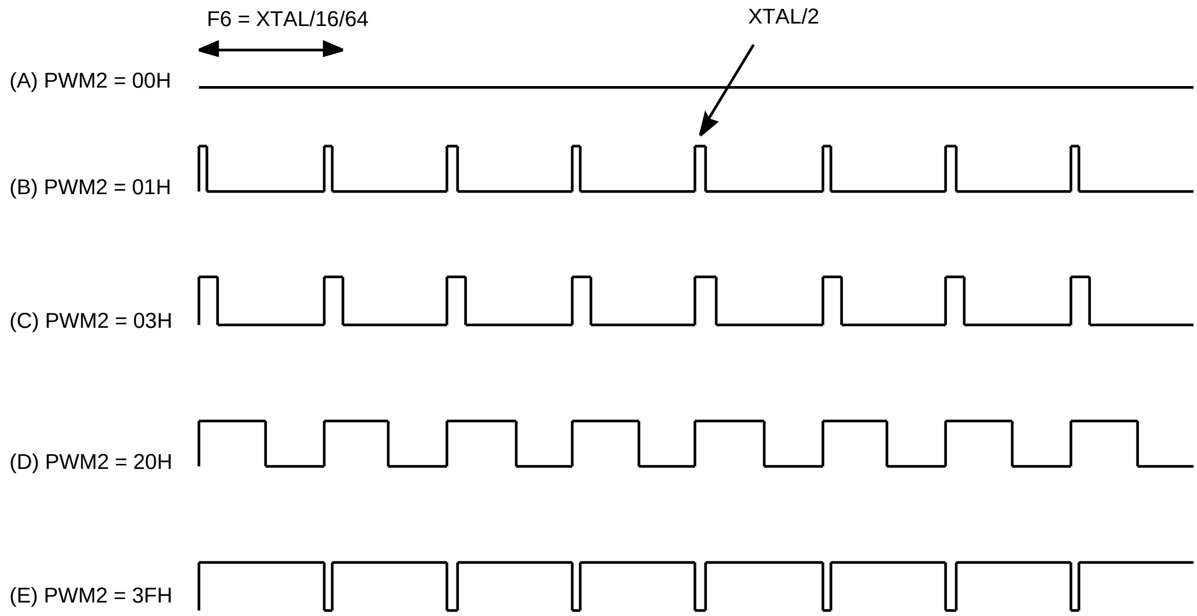


Figure 8-3. Pulse Width Modulator Timing Diagram, 6 Bit

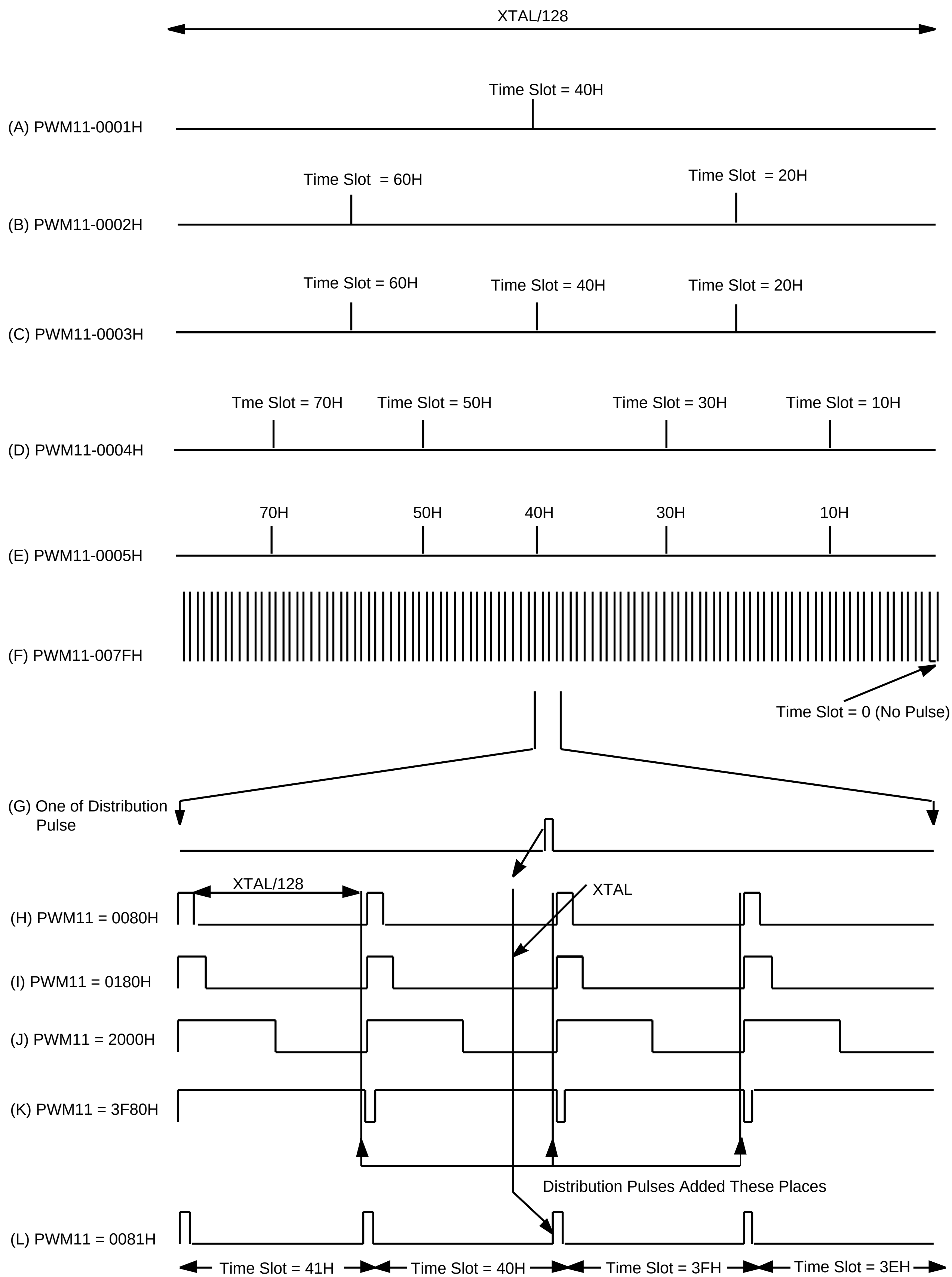


Figure 8-4. Pulse Width Modulator Timing Diagram, 14 Bit

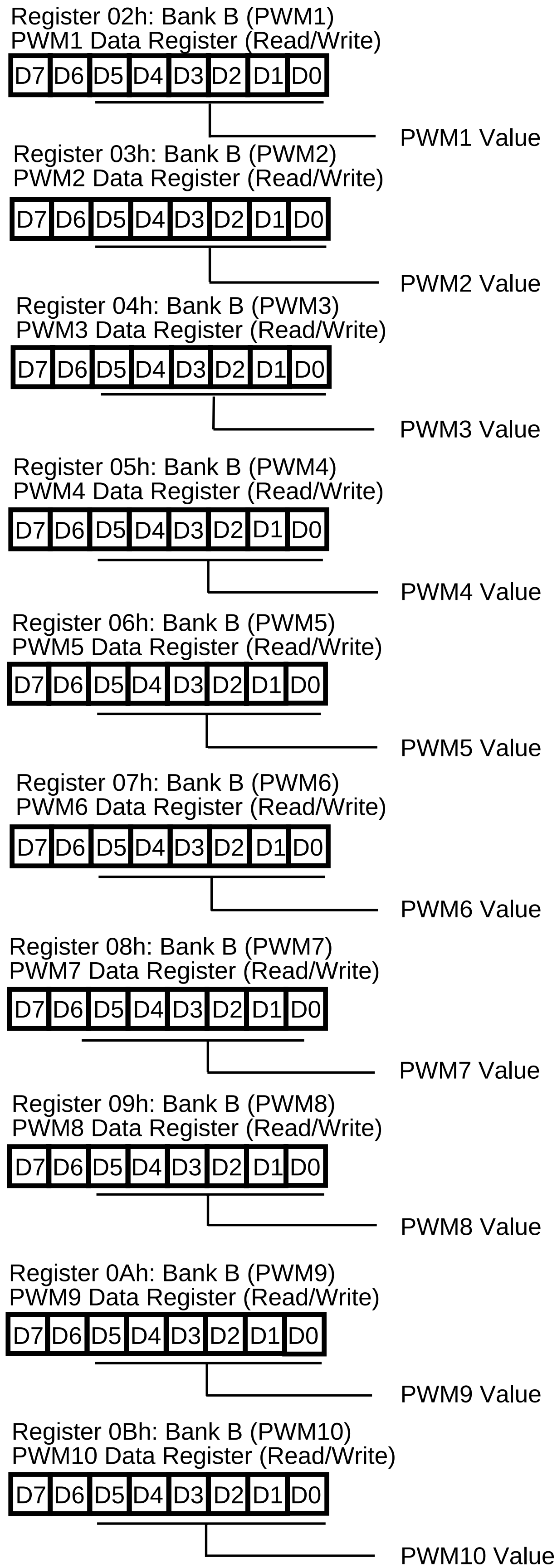


Figure 8-5. PWM1 through PWM10 Registers

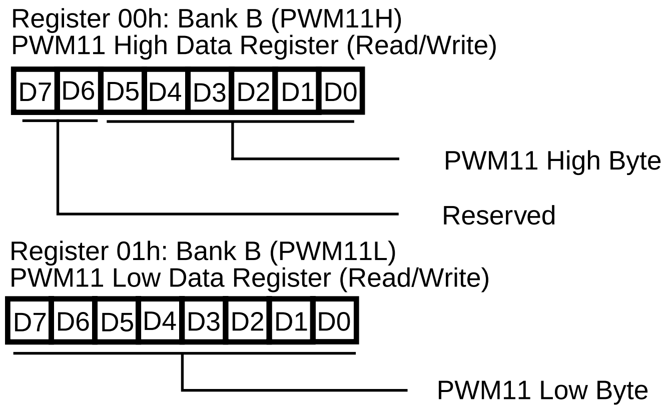


Figure 8-6. PWM11 Register

8.1.4 Digital/Analog Conversion via PWM

The DTC can generate square waves which have fixed periods but variable duty cycles. If such a signal is passed through an RC integrator, the output is a DC voltage proportional to the pulse width of the square wave. Cases A and B show fixed voltage samples while case C shows a varying voltage example.

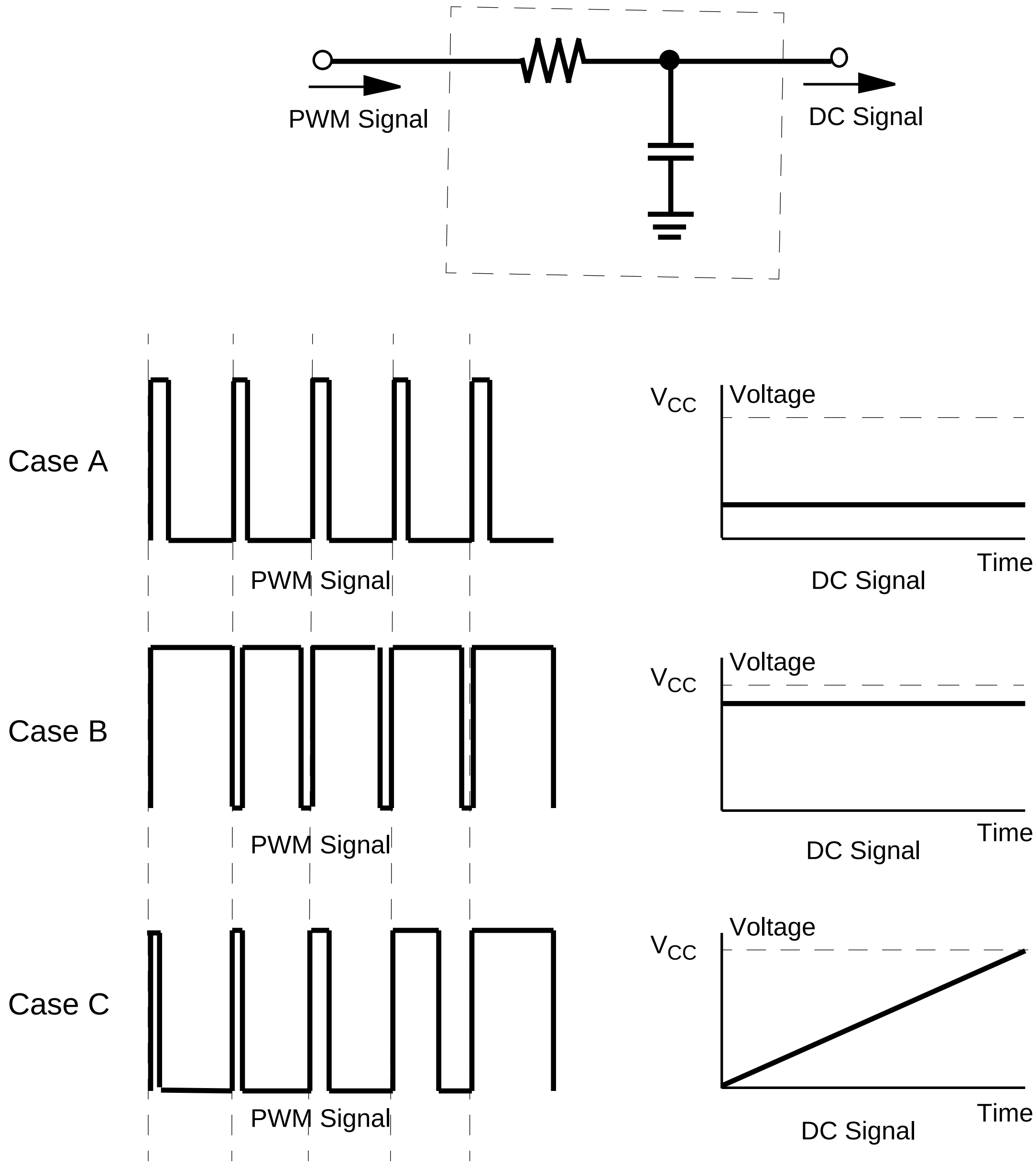


Figure 8-7. Analog Signals Generated from PWM Signals

APPENDIX A

PHILIPS I^{2}C SPECIFICATION

A.1 PHILIPS I^{2}C SPECIFICATION

This section comprises reference documentation for the I^{2}C bus. The material includes detailed information about application design, as well as a technical description of the bus itself.

The specification contained in this section provides the standard that the Z90230 family supports and to which application products should conform.

APPENDIX B

ANALOG PERIPHERALS

B.1 ANALOG-TO-DIGITAL CONVERTER

The Z90230 family is equipped with a 3-bit or 4-bit, depending on software configuration, flash analog-to-digital converter (ADC) with four multiplexed analog-input channels. There are two register addresses, one for 3-bit ADC (3ADC_DTA: 00h: Bank C) and the other for 4-bit ADC (4ADC_DTA: 01h: Bank F). Because no default is set, system software must configure the control register for the preferred ADC.

The converted 3-bit data is available as bits 0, 1, and 2 of the 3ADC data register (3ADC_DTA: 00h: Bank C).

The converted 4-bit data is available as bits 0, 1, 2, and 3 of the 4ADC data register (4ADC_DTA: 01h: Bank F).

Four input pins (P60/ADC3, P61/ADC2, P41/ADC1, and P62/ADC0) function as analog-input channels and as digital I/O ports. To support the analog function, the digital ports must be configured as analog through software. Analog/digital selection is controlled by bits D4 and D3 of the 3ADC Data Register for 3 bit and D5 and D4 of 4 ADC Data Register for 4 bit. If ADC Input Selection equals 00, ADC0 is selected; this value is the default following POR. If ADC Input Selection equals 01, ADC1 is selected. If ADC Input Selection equals 10, ADC2 is selected. If ADC Input Selection equals 11, ADC3 is selected.

Sampling occurs at one-eighth of an ADC-clock tick. On ADC-clock tick equals one-half, -third, or -quarter of a system-clock (SCLK) tick, as set by 3ADC_DTA(6,5) for 3 bit or 4ADC_DTA (7,6) for 4 bit. If ADC speed bits are set to 00, the ADC is not operative; this is the default value following POR. If these bits equal 01, ADC speed is based on one-half of a system-clock tick, SCLK/2. If these bits equal 10, ADC speed is based on one-third of a system-clock tick, SCLK/3. If these bits equal 11, ADC speed is based on one-quarter of a system-clock tick, SCLK/4.

B.1.1 3-Bit ADC Data Register

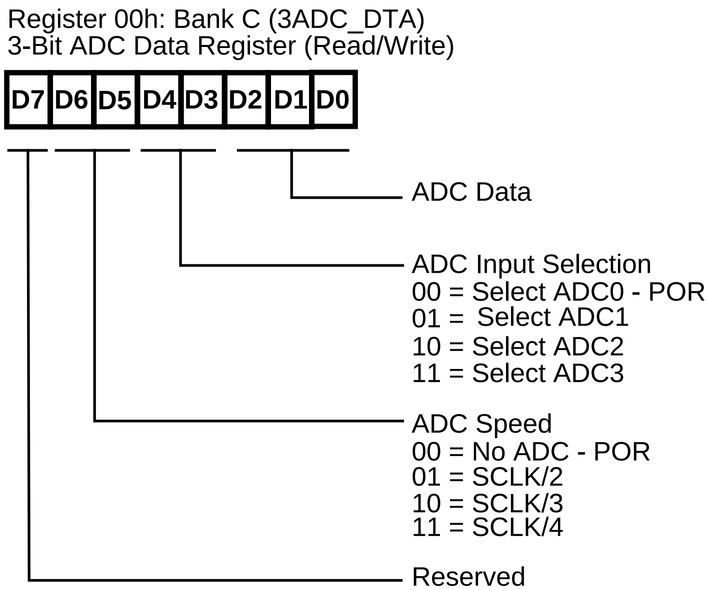


Figure B-1. 3-Bit ADC Data Register

B.1.2 4-Bit ADC Data Register

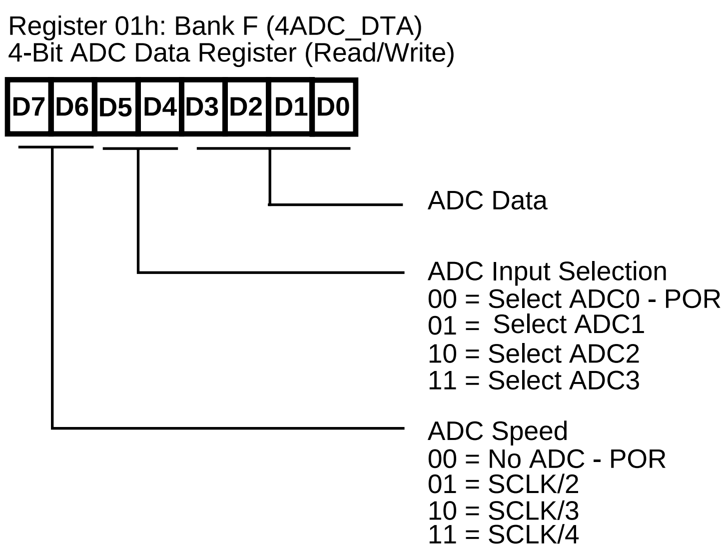


Figure B-2. 4-Bit ADC Data Register

P41 must be set to input mode for ADC 1 selection.

B.1.3 ADC Block Diagram

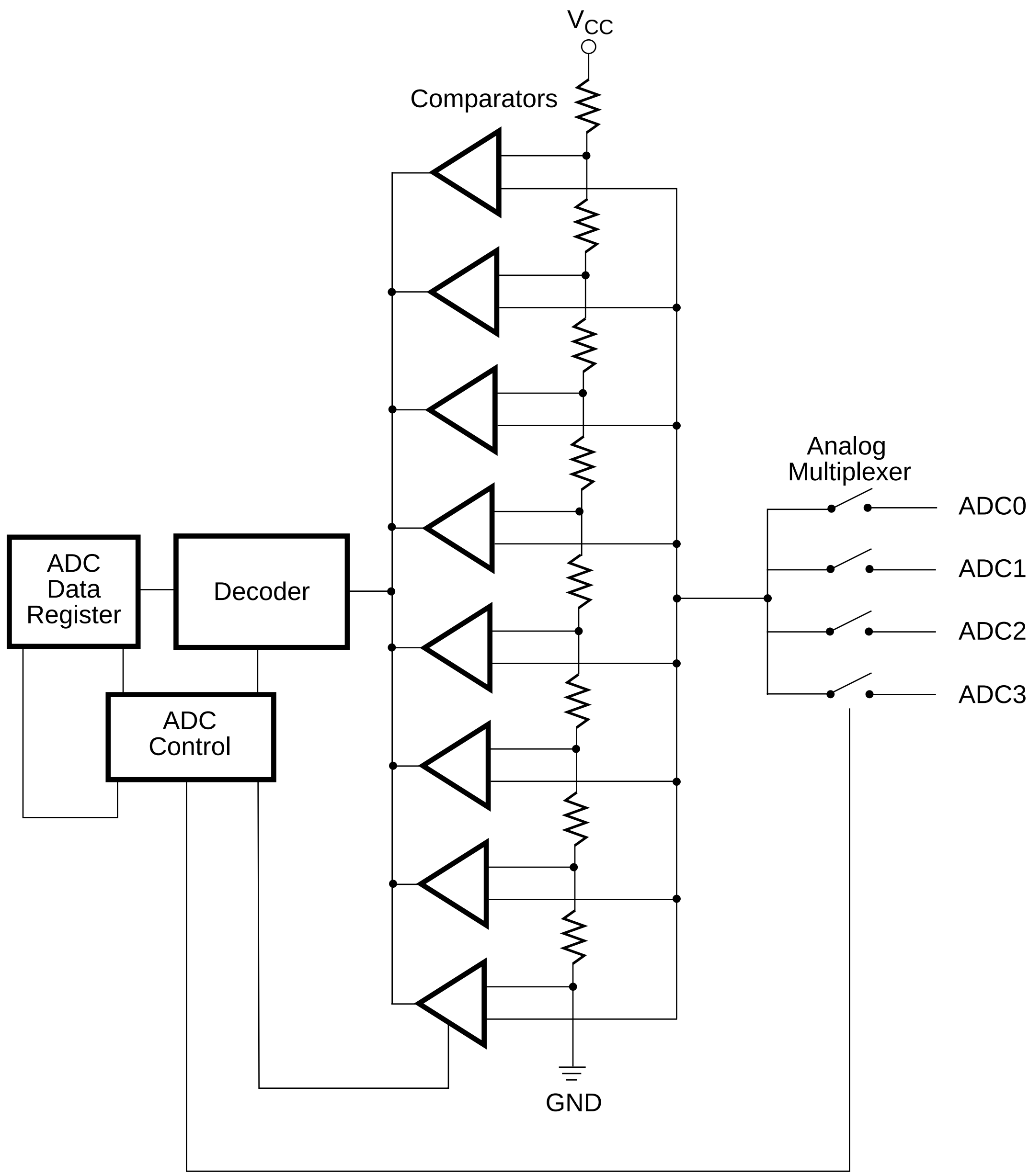


Figure B-3. ADC Block Diagram

APPENDIX C
SUPPORT PRODUCTS

C.1 Z90230 FAMILY SUPPORT PRODUCTS

The following development tools are available for use with the Z90230 family of DTCs.

C.1.1 ICEBOX Family In-Circuit Emulators

The Zilog ICEBOX product family of in-circuit emulators are interactive, Window-oriented development tools, featuring a real-time environment for emulation and debugging.

C.1.2 Z90219 Emulator (Z9021901ZEM)

Packages	Emulation	Programming
42-Pin SDIP	Z90233/Z90234	Z90231
124-Pin PGA	Z90219/Z90239	N/A

C.1.3 Z90219 Emulation Module (Z9020900TSC)

The Z90219 Emulation Module can be used like a One-Time Programmable (OTP) for plug-in emulation of the Z90230 family of devices in user target applications. It provides external EPROMS to simulate an OTP and can be used repeatedly. Its electrical characteristics are nearly identical with the OTP.

Packages	Emulation
42-Pin SDIP	Z90233/Z90234

C.1.4 Z89332 Evaluation Board (Z8933200ZCO)

The Z89332 Evaluation Board enables users to become familiar with the functions of the Z89300 and Z90230 family of devices in TV, VCR, and Cable Box environments. The board includes Z89 OTP, pre-programmed, with sample code to demonstrate the Applications Programming Interface (API).

Packages	Supported Devices
42-Pin SDIP	Z89331/Z89332
42-Pin SDIP	Z90233/Z90234

C.1.5 ICEBOX/HP Logic Analyzer Adapter Board (Z89C0000ZHP)

The ICEBOXHP Logic Analyzer Adapter Board provides users of the HP Logic Analyzer (165XX Series) with real-time trace capabilities for Zilog ICEBOX Emulators. Captured code can be disassembled, providing a complete listing of program flow in native assembly language on the analyzer screen.

Supported Devices

Z89301	Z89331	Z89346	Z89239
Z89313	Z89332	Z89300	Z89319
Z89341	Z90231	Z90239	

C.1.6 Zilog Macro Cross Assembler (ZMASM0W0ZAS)

Zilog's Macro Cross Assembler (ZMASM), is a powerful and full-featured relocatable assembler that enhances programmer productivity. It is designed as a perfect match for use with the Zilog ICEBOX line of in-circuit emulators and also with Zilog’s evaluation boards, but is still compatible with other vendor’s products as well.

ZMASM processes assembly language source code written for a supported device (target processor) and translates it into the binary code that the processor can execute. ZMASM can also provide source level debug information in the object file.

ZMASM has a graphical user based project front-end interface that efficiently manages large numbers of source files so only the minimum number of required files are reassembled when source code changes are made. ZMASM also has a command line interface mode available.

C.1.7 ZMASM Supported Cores/ Devices

Processor Cores

Z8-based

Z89C00-based (DSP)

Target Processor Devices

Z86C47/E47	Z89313	Z90102	Z90231
Z89300	Z89319	Z90103	Z90233
Z89301	Z89331	Z90104	Z90341
Z89302	Z89332	Z90211	Z90346

APPENDIX D
REGISTERS

D.1 REGISTERS

This section serves as a quick reference to the Z90230 data registers. The following registers are contained in this appendix:

Description	Page
Expanded Register File	D-2
Register and Expanded Register File Map	D-3
OSD Control Register	D-4
Vertical Position Register	D-4
Display Attribute Register	D-4
Display Attribute Register	D-5
Row Space Register	D-5
Fade Position Register 1	D-6
Fade Position Register 2	D-6
Second Color Control Register	D-6
Second Color Register	D-6
Color Palette 0	D-7
Color Palette 1	D-7
Color Palette 2	D-7
Color Palette 3	D-7
Color Palette 4	D-8
Color Palette 5	D-8
Color Palette 6	D-8
PWM11 Register	D-8
PWM1 through PWM10 Registers	D-9
Row Attribute Register	D-10
Port 5 Data Register	D-10
PWM Mode Register	D-11
Port 5 Direction Control Register	D-12
3-Bit ADC Data Register	D-12
Timer Control Register 0	D-13
Timer Control Register 1	D-13
IR Capture Register 0	D-13
IR Capture Register 1	D-14
Port 4 Data Register	D-14
Port 4 Direction Control Register	D-15
HV Interrupt Status Register	D-16
Port 4 Pin-Out Selection Register	D-16
Color Index Register	D-17
Master I^2C Data Register	D-17
Master I^2C Command Register	D-17
Master I^2C Control Register	D-18
Port Configuration Register	D-19
4-Bit ADC Data Register	D-19
Port 6 Direction Control Register	D-20
Port 6 Data Register	D-21
Mesh Column Start Register	D-21
Mesh Column End Register	D-22
Mesh Row Enable Register	D-22
Mesh Control Register	D-23
Stop-Mode Recovery Register	D-24
Watch-Dog Timer Mode Register	D-24
Stack Pointer Low Register	D-25
Stack Pointer High Register	D-25
Register Pointer	D-25
Flag Register	D-25
Interrupt Mask Register	D-26
Interrupt Request Register	D-26
Interrupt Priority Register	D-27
Port 2 Control Register	D-27
Port 2 Mode Register	D-28
Prescaler 0 Register	D-28
Counter/Timer 0 Register	D-29
Prescaler 1 Register	D-29
Counter/Timer 1 Register	D-29
Timer Mode Register	D-30
Port 2 Data Register	D-30

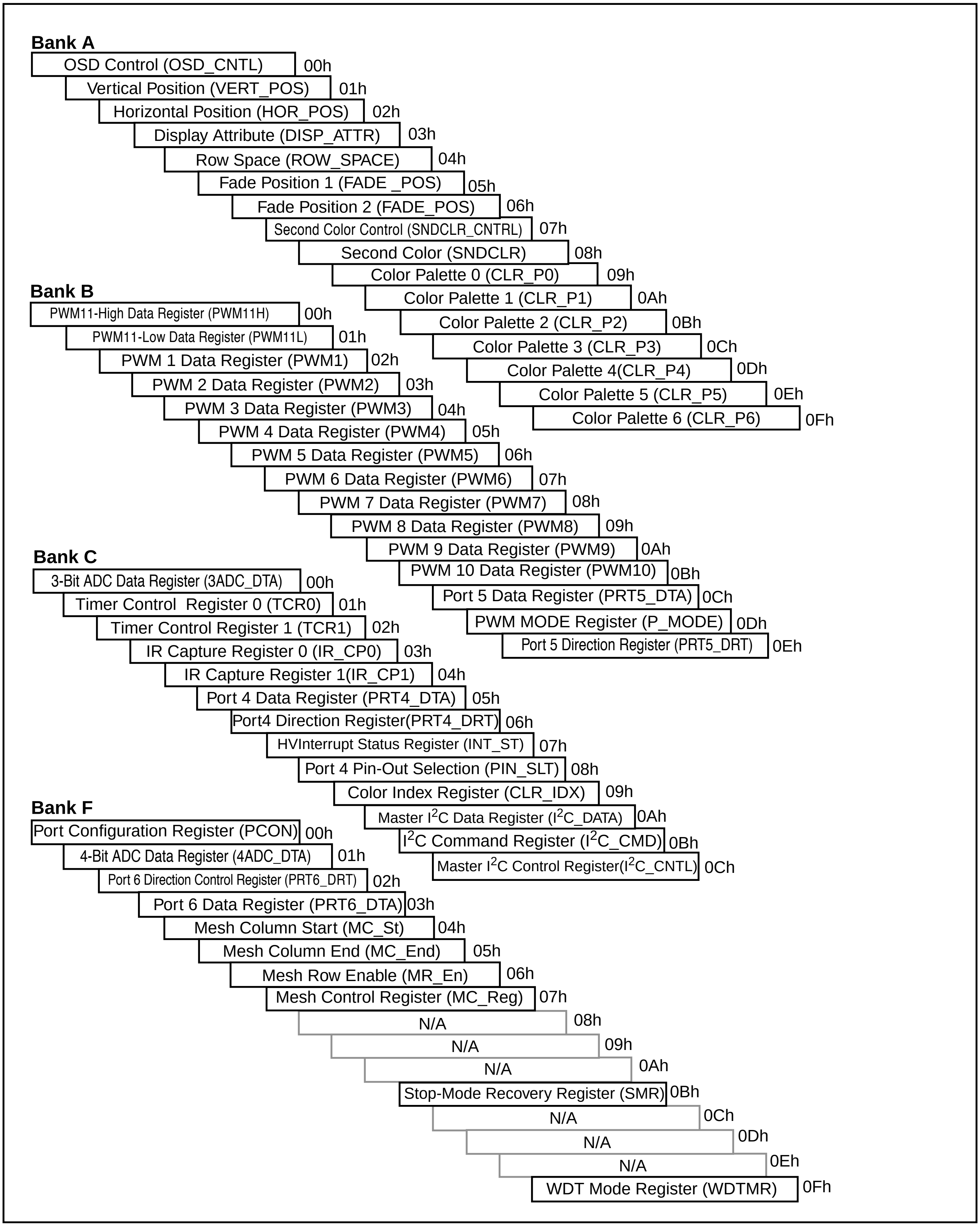
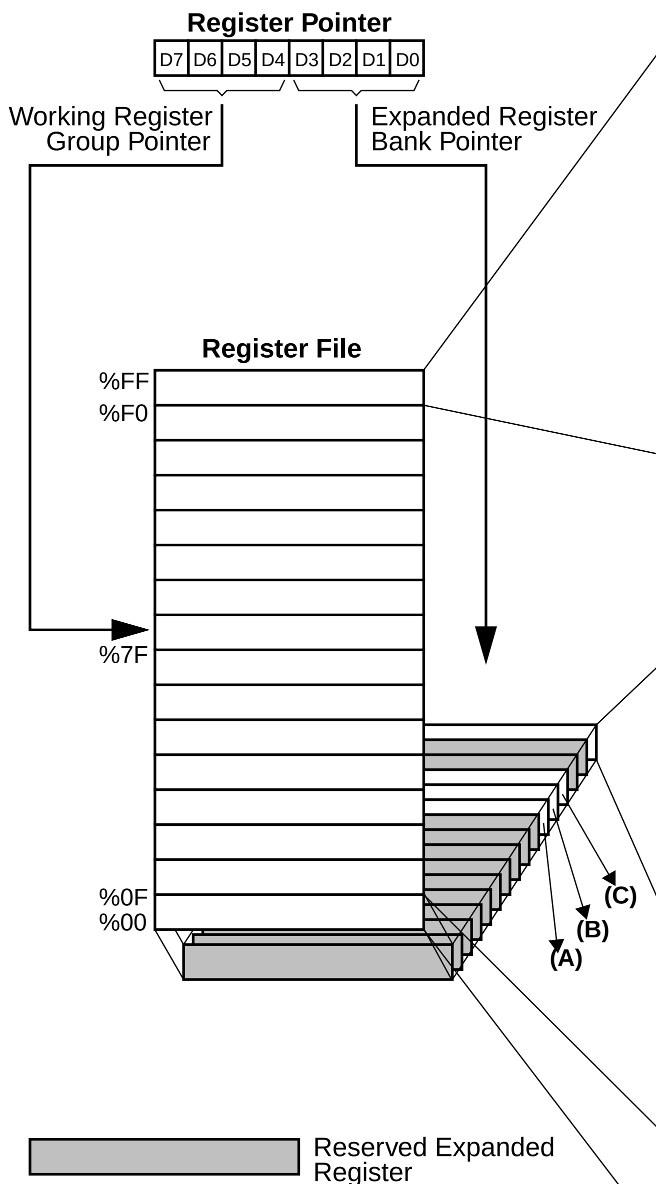


Figure D-1. Expanded Register File



Register		Reset Condition D7	D6	D5	D4	D3	D2	D1	D0
%FF	SPL	U	U	U	U	U	U	U	U
%FE	SPH	U	U	U	U	U	U	U	U
%FD	RP	0	0	0	0	0	0	0	0
%FC	FLAGS	U	U	U	U	U	U	U	U
%FB	IMR	0	U	U	U	U	U	U	U
%FA	IRQ	0	0	0	0	0	0	0	0
%F9	IPR	U	U	U	U	U	U	U	U
%F8	P01M	1	1	1	1	1	1	1	1
%F7	P2CNTL	0	0	0	0	0	0	0	1
%F6	P2M	1	1	1	1	1	1	1	1
%F5	PRE0	U	U	U	U	U	U	U	0
%F4	T0	U	U	U	U	U	U	U	U
%F3	PRE1	U	U	U	U	U	U	0	0
%F2	T1	U	U	U	U	U	U	U	U
%F1	TMR	0	0	0	0	0	0	0	0
%F0	Reserved								

Expanded Register Bank (F)

Register		Reset Condition							
%(F)0F	WDTMR	U	U	U	0	1	1	0	1
%(F)0E	Reserved								
%(F)0D	Reserved								
%(F)0C	Reserved								
%(F)0B	SMR	0	0	1	0	0	0	0	0
%(F)0A	Reserved								
%(F)09	Reserved								
%(F)08	Reserved								
%(F)07	MC_Reg	0	0	0	0	0	0	0	0
%(F)06	MR_En	0	0	0	0	0	0	0	0
%(F)05	MC_End	U	U	U	0	0	0	0	0
%(F)04	MC_St	U	U	U	0	0	0	0	0
%(F)03	PRT6_DTA	U	U	U	U	1	1	1	1
%(F)02	PRT6_DRT	1	1	1	1	1	1	1	1
%(F)01	4ADC_DTA	0	0	0	0	U	U	U	U
%(F)00	PCON	1	1	1	1	1	1	1	0

Register		Reset Condition							
%(0)03	Reserved								
%(0)02	P2	U	U	U	U	U	U	U	U
%(0)01	Reserved								
%(0)00	Reserved								

Figure D-2. Register and Expanded Register File Map

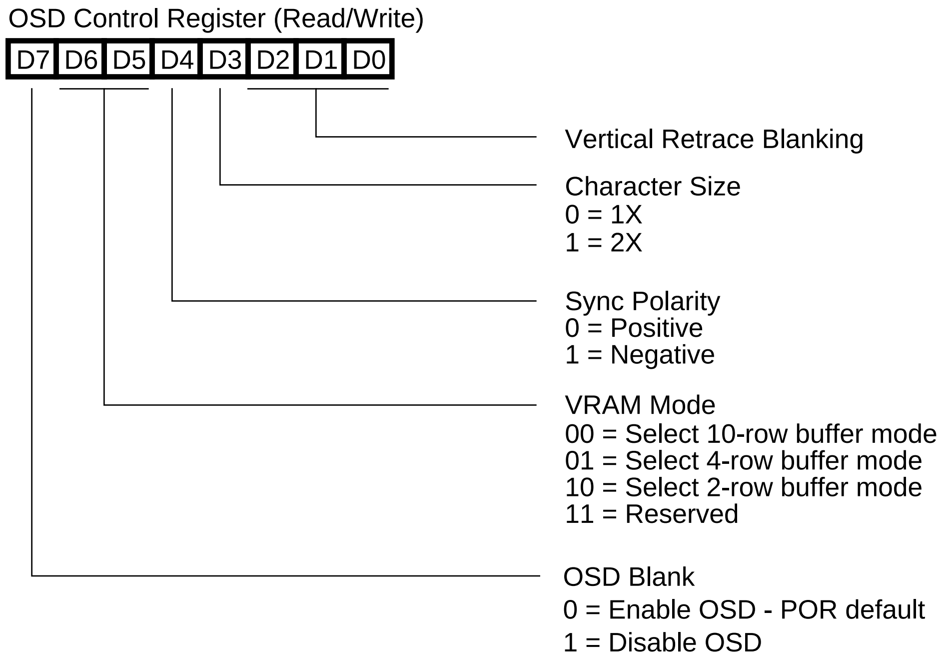


Figure D-3. OSD Control Register

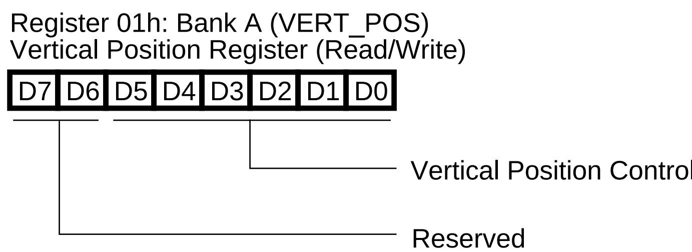


Figure D-4. Vertical Position Register

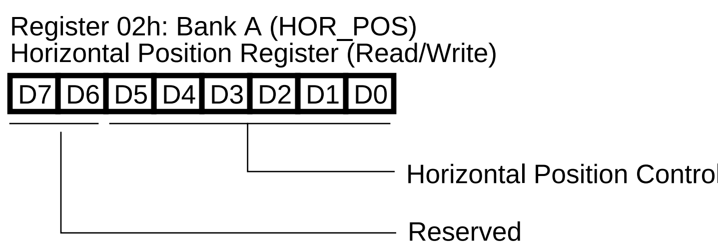


Figure D-5. Horizontal Position Register

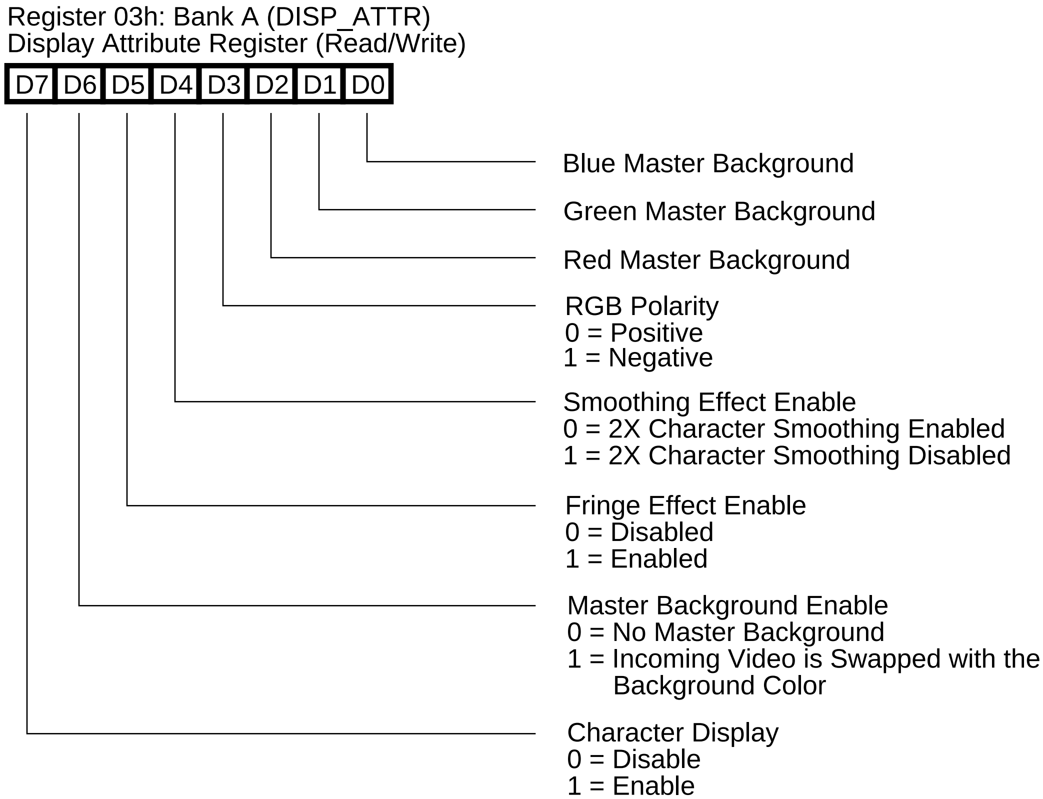


Figure D-6. Display Attribute Register

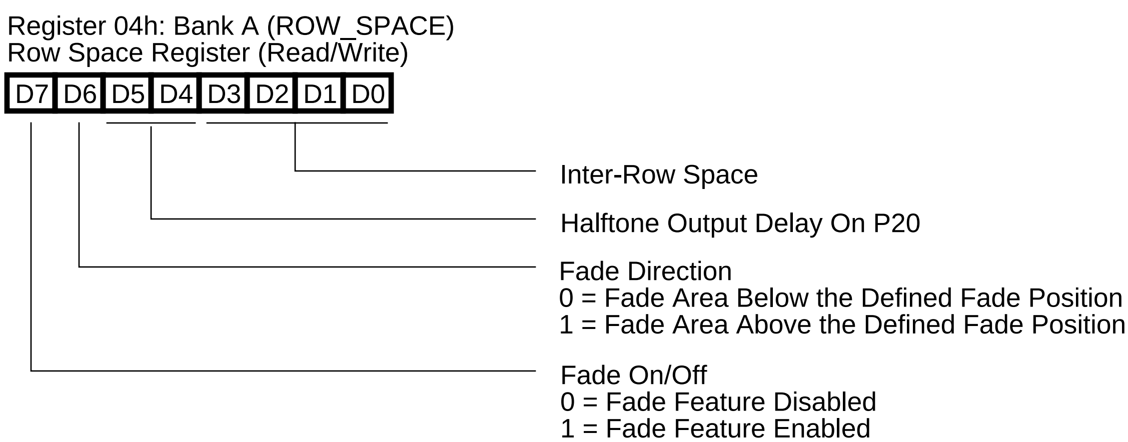


Figure D-7. Row Space Register

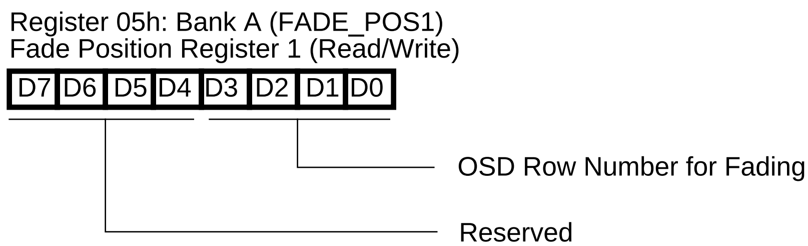


Figure D-8. Fade Position Register 1

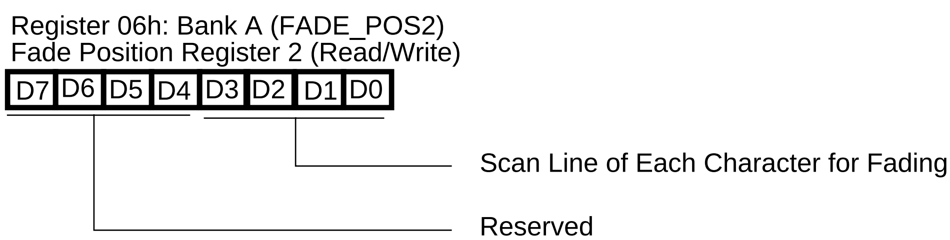


Figure D-9. Fade Position Register 2

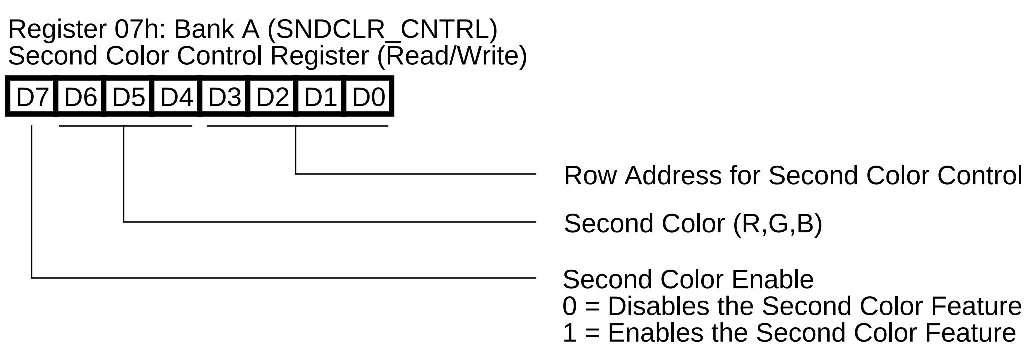


Figure D-10. Second Color Control Register

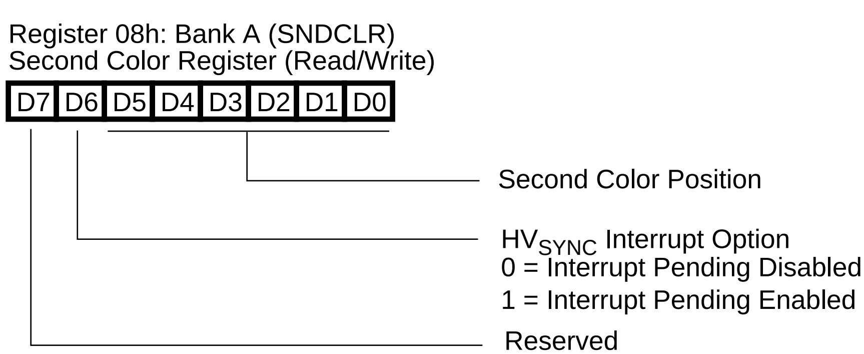


Figure D-11. Second Color Register

Register 09h: Bank A (CLR_P0)
Color Palette 0 (Read/Write)

Figure D-12. Color Palette 0

Register 0Ah: Bank A (CLR_P1)
Color Palette 1 (Read/Write)

Figure D-13. Color Palette 1

Register 0Bh: Bank A (CLR_P2)
Color Palette 2 (Read/Write)

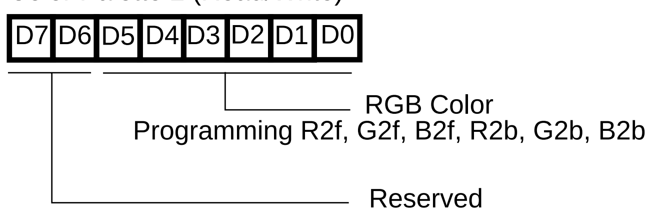


Figure D-14. Color Palette 2

Register 0Ch: Bank A (CLR_P3)
Color Palette 3 (Read/Write)

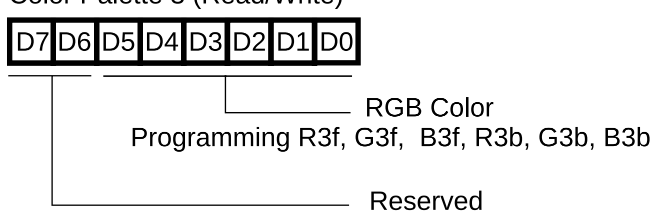


Figure D-15. Color Palette 3

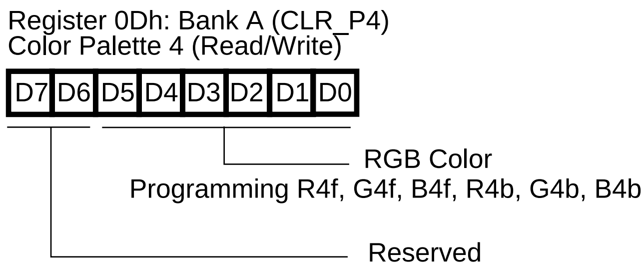


Figure D-16. Color Palette 4

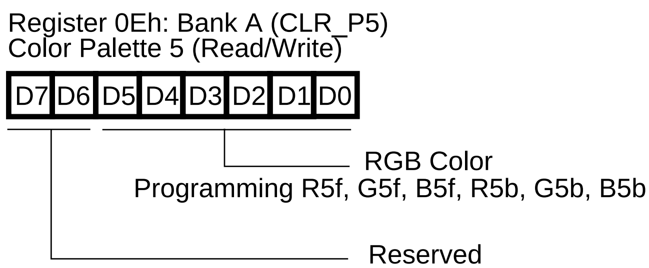


Figure D-17. Color Palette 5

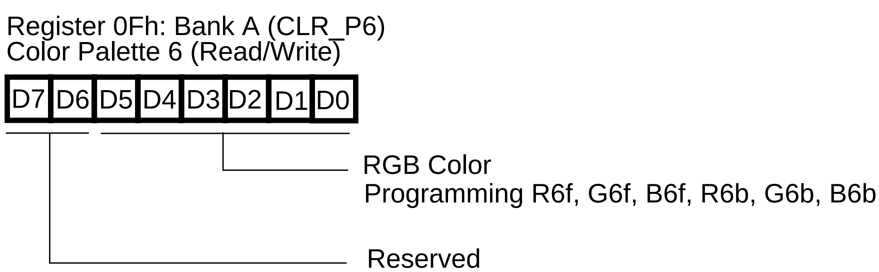


Figure D-18. Color Palette 6

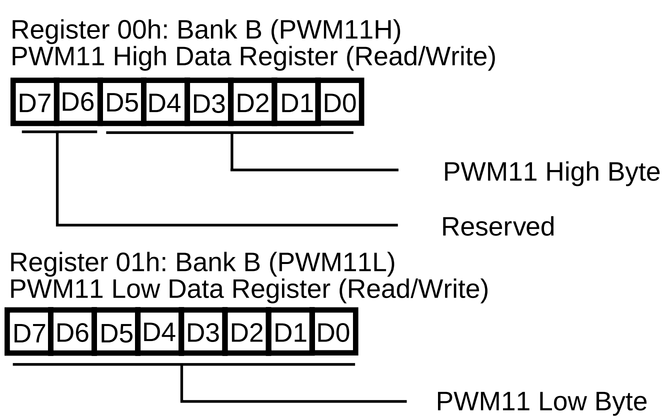


Figure D-19. PWM11 Register

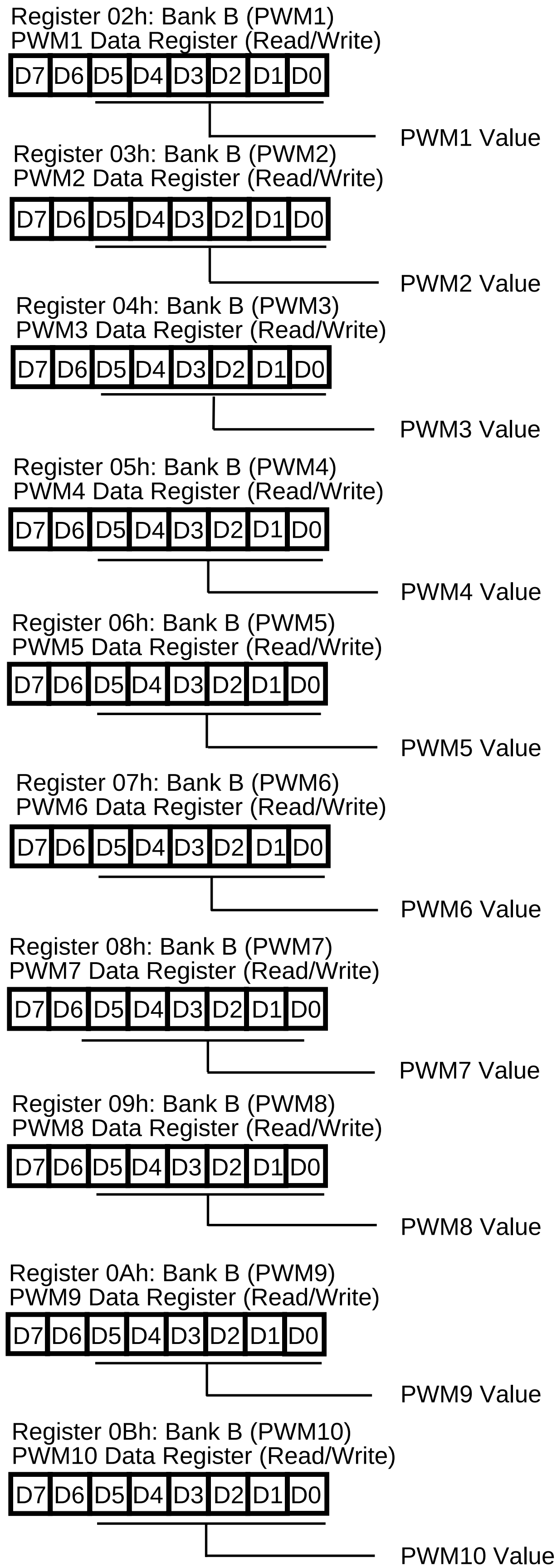


Figure D-20. PWM1 through PWM10 Registers

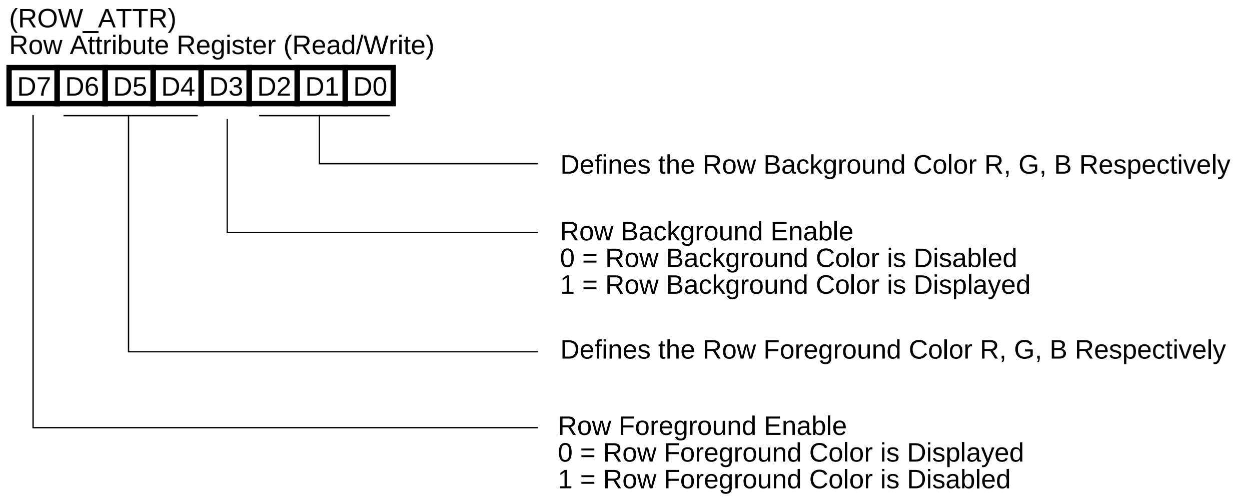


Figure D-21. Row Attribute Register

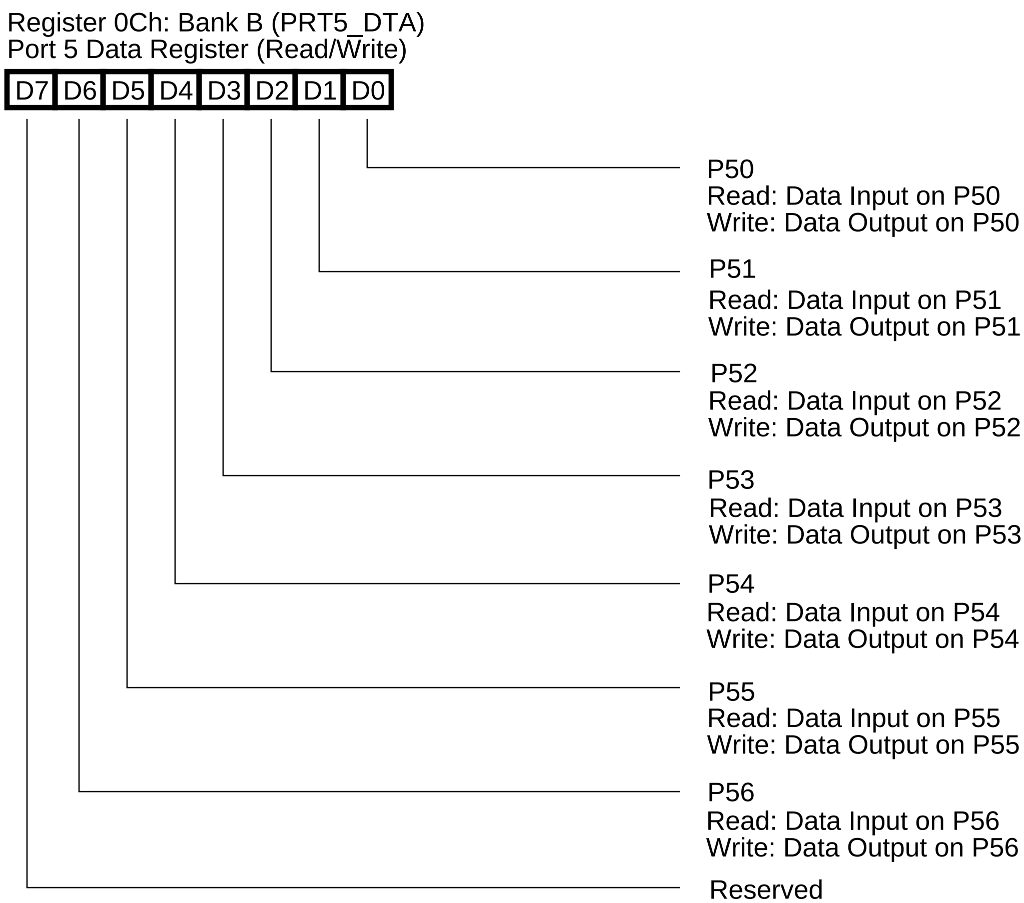


Figure D-22. Port 5 Data Register

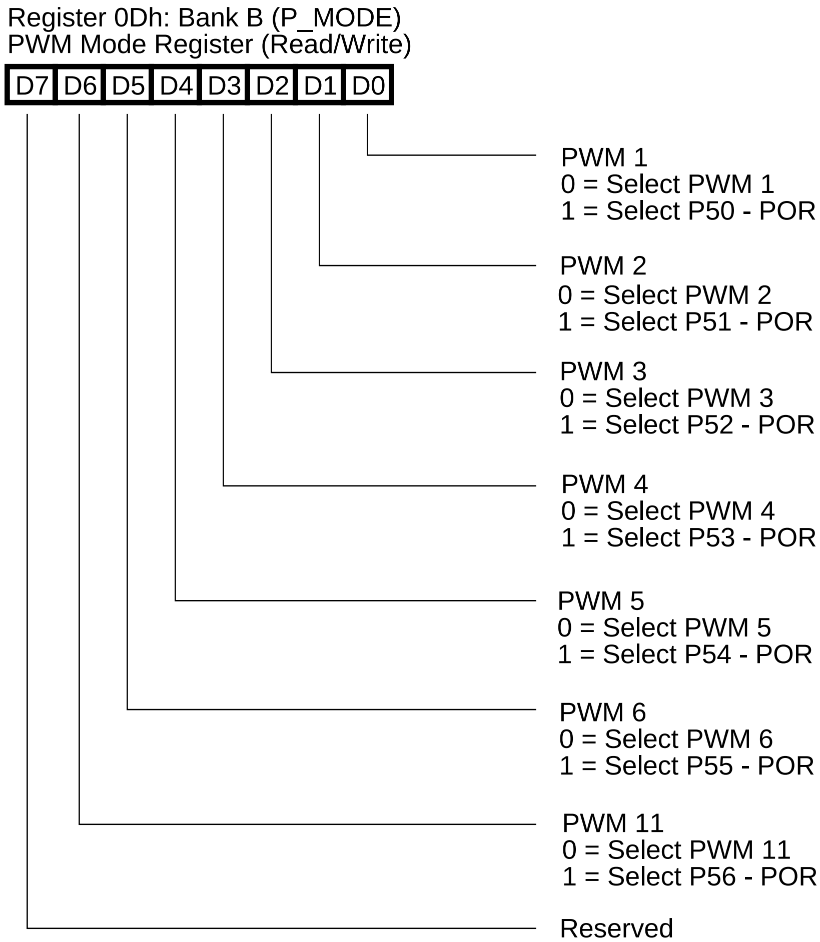


Figure D-23. PWM Mode Register

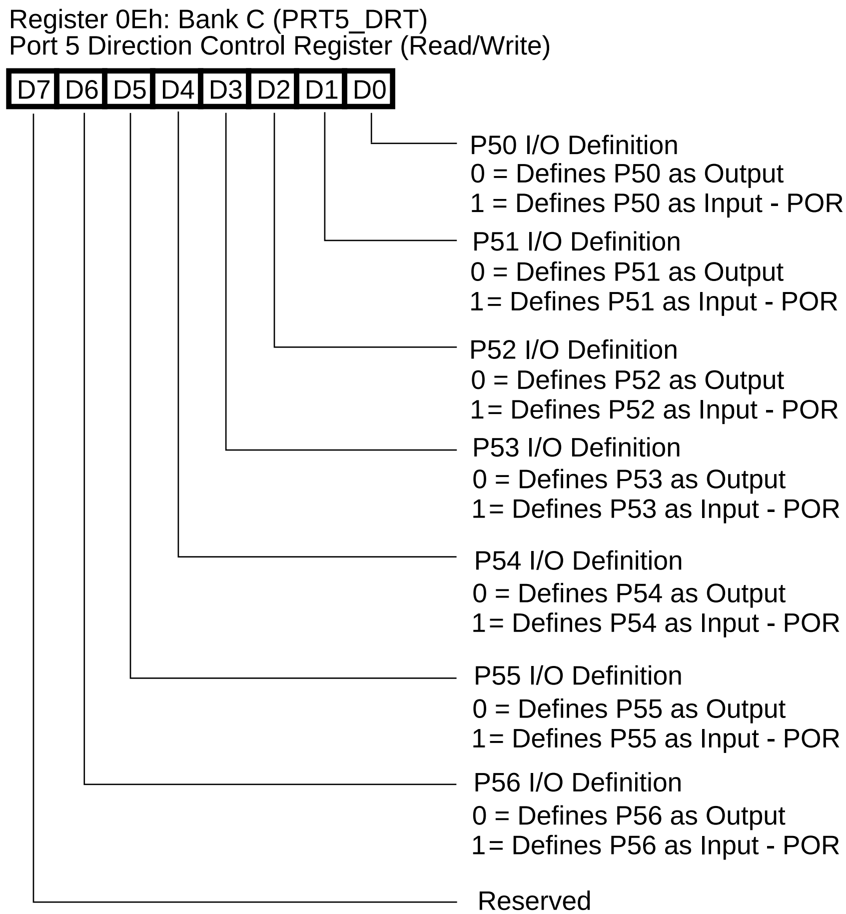


Figure D-24. Port 5 Direction Control Register

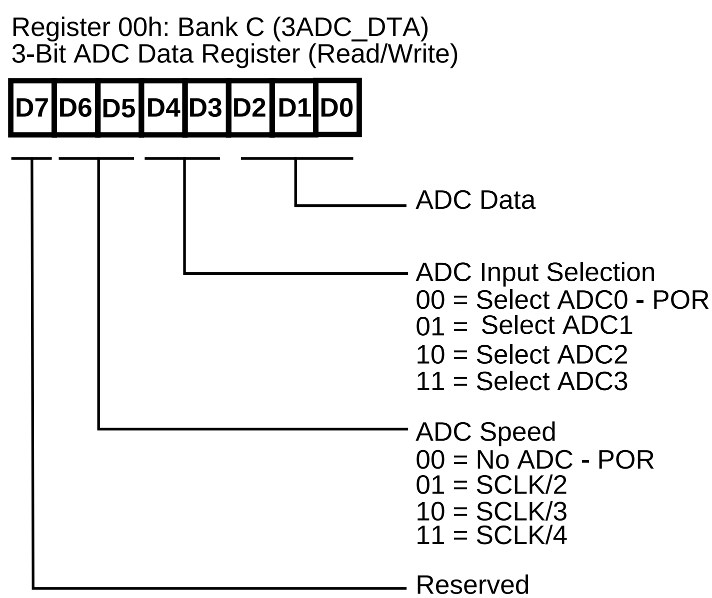


Figure D-25. 3-Bit ADC Data Register

Register 01h: Bank C (TCR0)
Timer Control Register 0 (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

D0: Tout_CAP
Read 0 = No Timeout of the Capture Timer
1 = Timeout of the Capture Timer
Write 0 = No Effect
1 = Reset Flag

D1: CAPint_f
Read 0 = No Falling Edge is Captured
1 = Falling Edge is Captured
Write 0 = No Effect
1 = Reset Flag

D2: CAPint_r
Read 0 = No Rising Edge is Captured
1 = Rising Edge is Captured
Write 0 = No Effect
1 = Reset Flag

D7–D3: Reserved

Figure D-26. Timer Control Register 0

Register 02h: Bank C (TCR1)
Timer Control Register 1 (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

D1–D0: CAP Speed
00 = SCLK/32
01 = SCLK/4
10 = SCLK/8
11 = SCLK/16

D3–D2: CAP Glitch
00 = Glitch Filter Disabled
01 = <2SCLK Filtered Out
10 = <8SCLK Filtered Out
11 = <16SCLK Filtered Out

D5–D4: CAP Edge
00 = No capture
01 = Capture on Rising Edge Only
10 = Capture on Falling Edge Only
11 = Capture on Both Edges

D6: CAP Halt
0 = Capture Timer Running
1 = Capture Timer Halted

D7: Reserved

Figure D-27. Timer Control Register 1

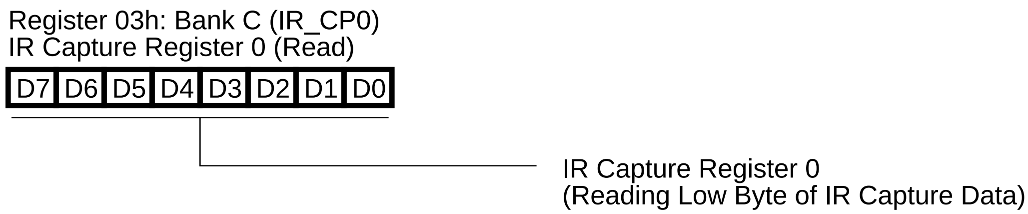


Figure D-28. IR Capture Register 0

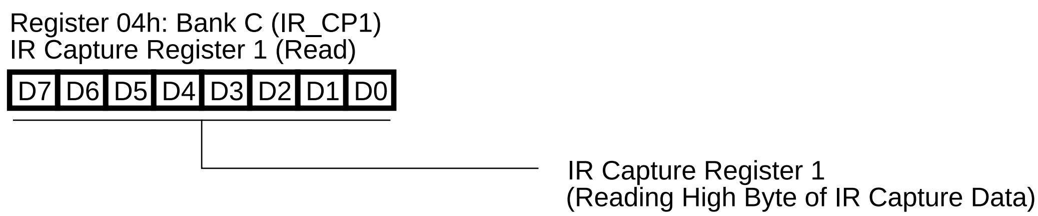


Figure D-29. IR Capture Register 1

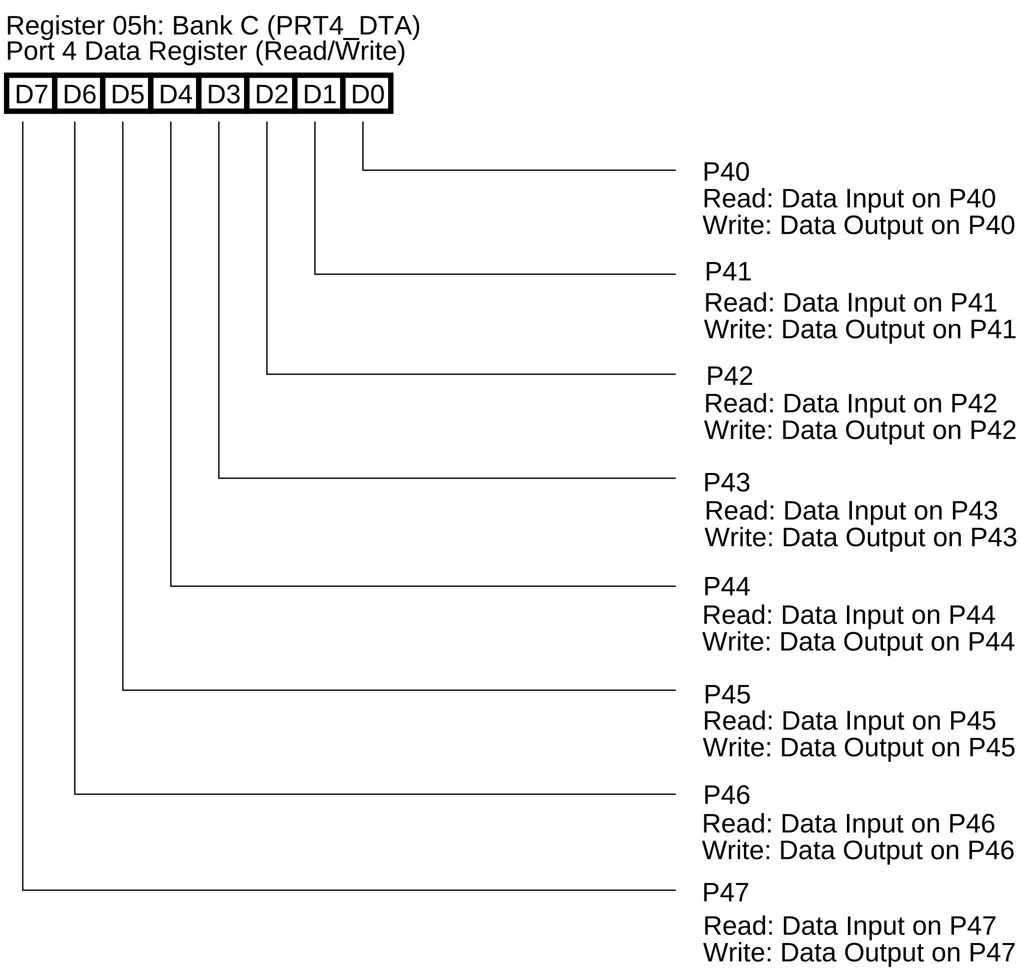


Figure D-30. Port 4 Data Register

Register 06h: Bank C (PRT4_DRT)
Port 4 Direction Control Register (Read/Write)

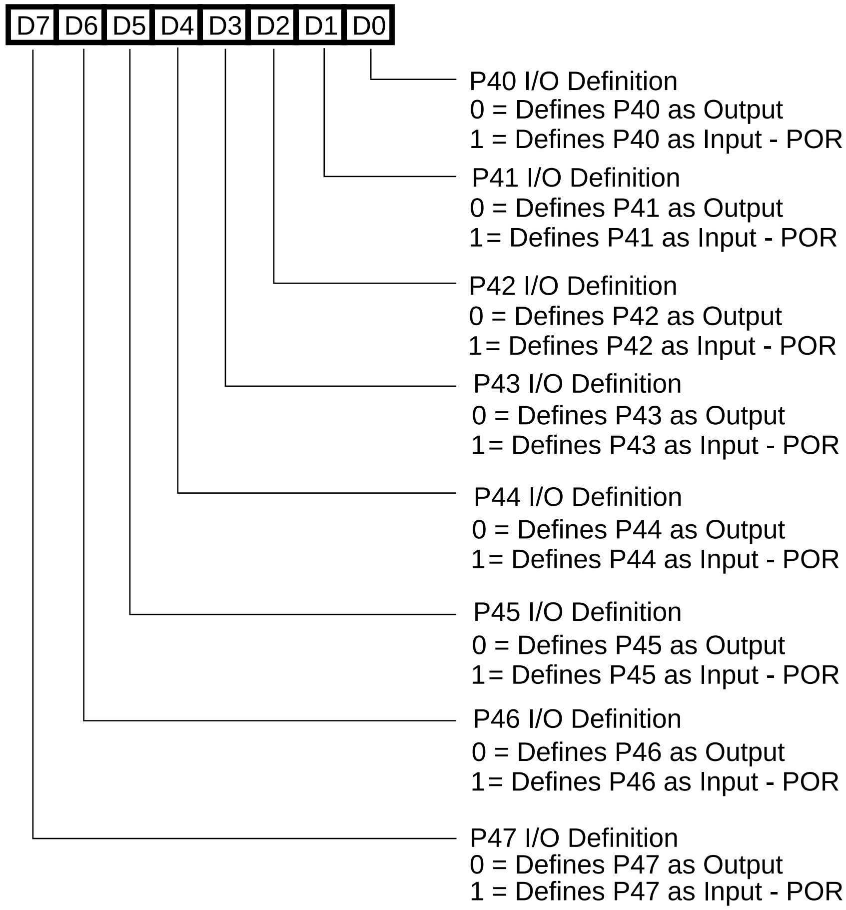


Figure D-31. Port 4 Direction Control Register

Register 07h: Bank C (INT_ST)
HV Interrupt Status Register (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

D0: Horizontal Interrupt
Read 0 = No Horizontal Interrupt
1 = Horizontal Interrupt
Write 0 = No Effect
1 = Reset Horizontal Interrupt Flag

D1: Vertical Interrupt
Read 0 = No Vertical Interrupt
1 = Vertical Interrupt
Write 0 = No Effect
1 = Reset Vertical Interrupt Flag

D2: Horizontal Interrupt Enable
0 = No Horizontal Interrupt
1 = Enable Horizontal Interrupt

D3: Palette Mode
0 = Normal Mode
1 = Color Palette Mode

D6–D4: Fringe Color
(Defines Fringe Color RGB)

D7: Fringe Color Selection
0 = Select Character Background Color
1 = Select Fringe Color RGB

Figure D-32. HV Interrupt Status Register

Register 08h: Bank C (PIN_SLT)
Port 4 Pin-Out Selection Register (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

D0: Reserved

D1: Reserved

D2: P44/PWM7
0 = Selects PWM7
1 = Selects P44 - POR

D3: P45/PWM8
0 = Selects PWM8
1 = Selects P45 - POR

D4: P46/PWM9
0 = Selects PWM9
1 = Selects P46 - POR

D5: P47/PWM10
0 = Selects PWM10
1 = Selects P47 - POR

D7–D6: Reserved

Figure D-33. Port 4 Pin-Out Selection Register

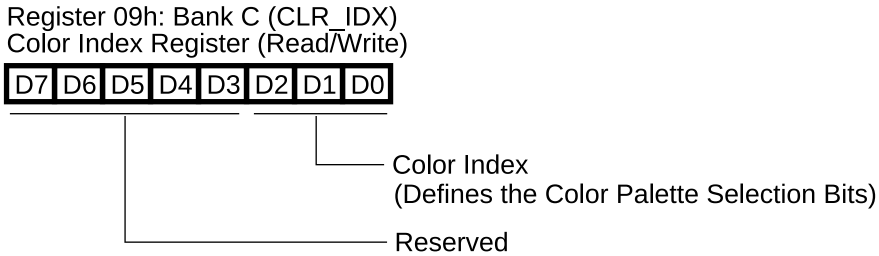


Figure D-34. Color Index Register

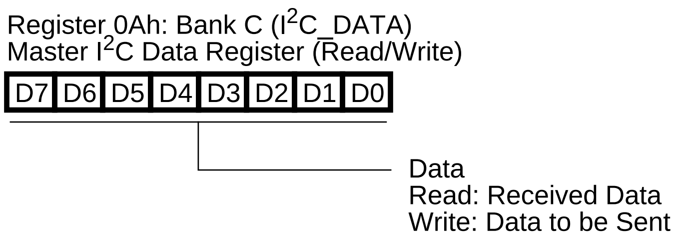


Figure D-35. Master I^{2}C Data Register

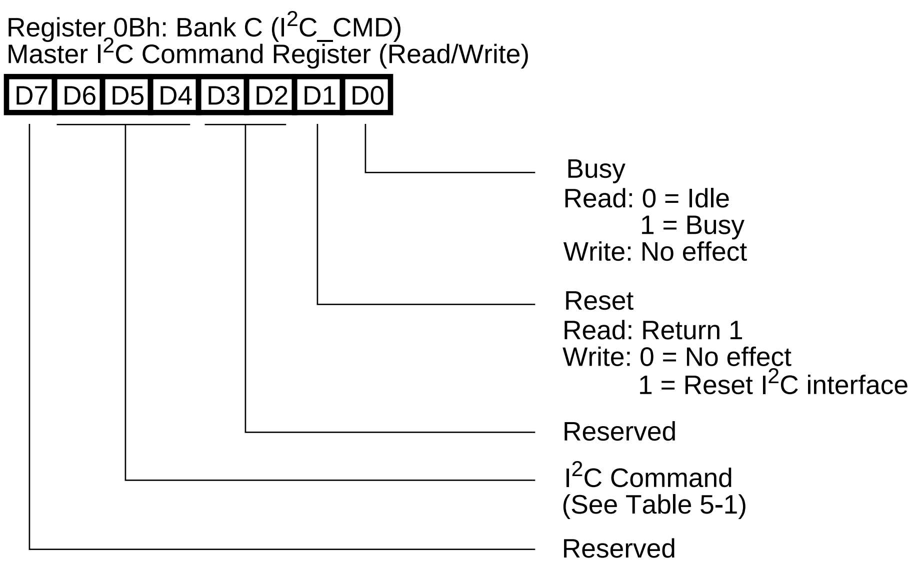


Figure D-36. Master I^{2}C Command Register

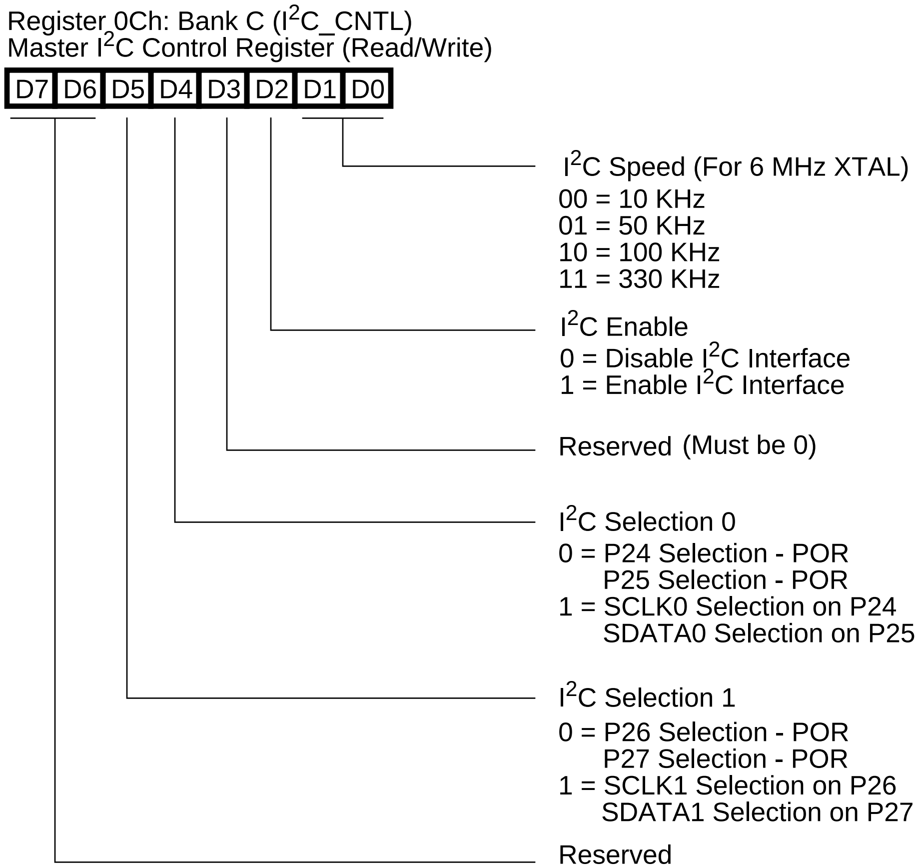


Figure D-37. Master I^2C Control Register

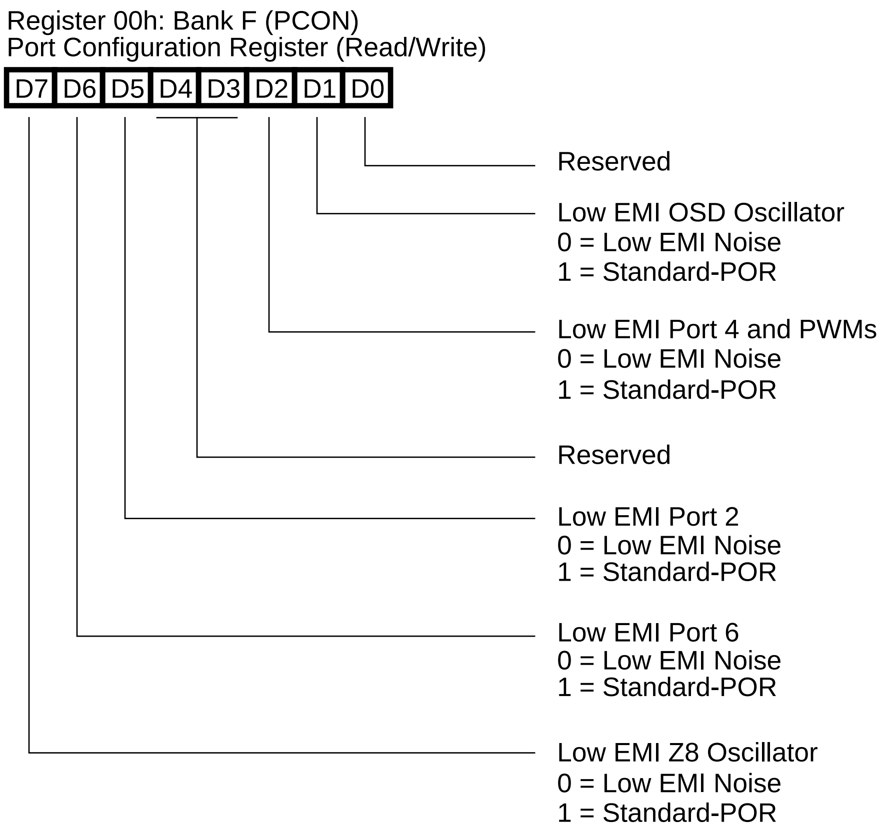


Figure D-38. Port Configuration Register

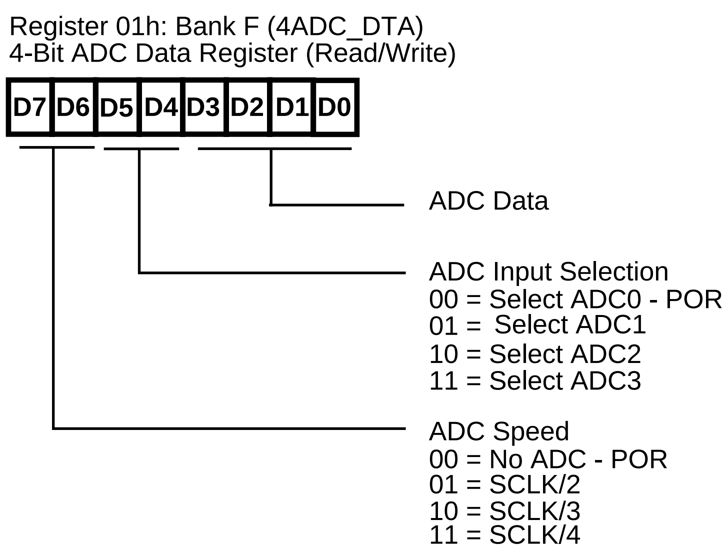


Figure D-39. 4-Bit ADC Data Register

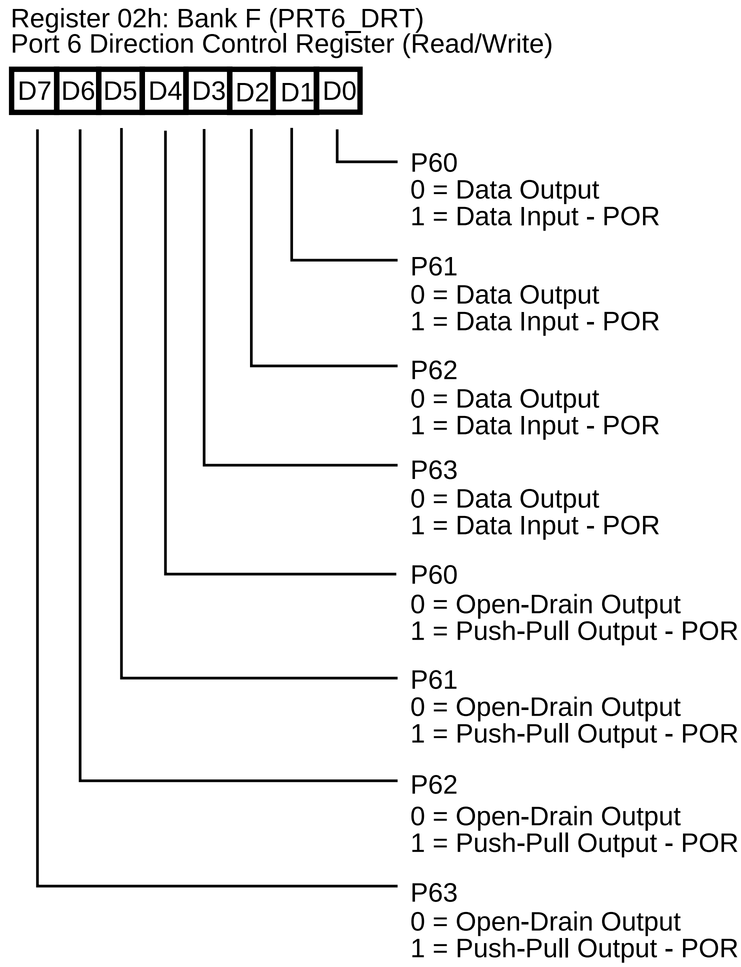


Figure D-40. Port 6 Direction Control Register

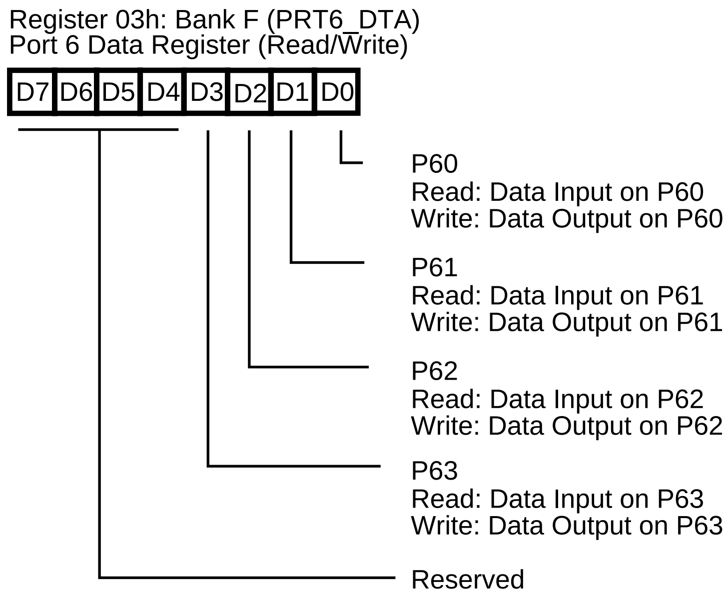


Figure D-41. Port 6 Data Register

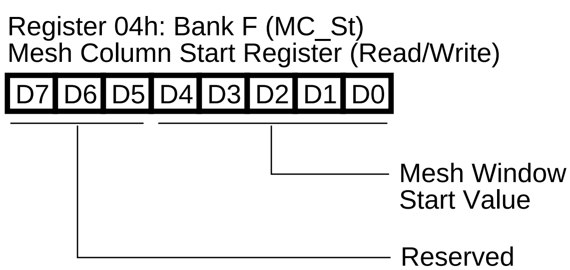


Figure D-42. Mesh Column Start Register

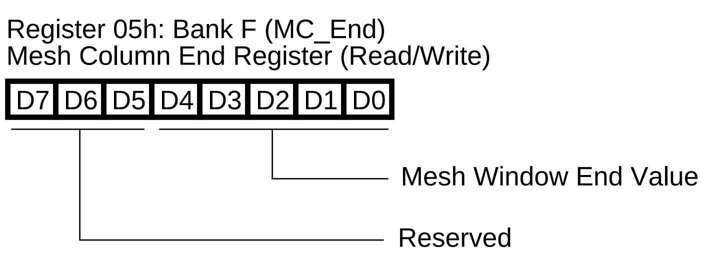


Figure D-43. Mesh Column End Register

Register 06h: Bank F (MR_En)
Mesh Row Enable Register (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

Mesh Window Row
0 = No mesh OSD for Next Row
1 = Mesh OSD for Next Row

Reserved (Must be 0)

Reserved (Must be 0)

Foreground Character for Halftone
0 = Not Included
1 = Included

V_{BLANK} Delay
0000 = No delay
0001 = Delay by 0.5 Dot-Clock Period
0010 = Delay by 1.0 Dot-Clock Period
0011 = Delay by 1.5 Dot-Clock Period
0100 = Delay by 2.0 Dot-Clock Period
0101 = Delay by 2.5 Dot-Clock Period
0110 = Delay by 3.0 Dot-Clock Period
0111 = Delay by 3.5 Dot-Clock Period
1000 = Delay by 4.0 Dot-Clock Period
1001 = Delay by 4.5 Dot-Clock Period
1010 = Delay by 5.0 Dot-Clock Period
1011 = Delay by 5.5 Dot-Clock Period
1100 = Delay by 6.0 Dot-Clock Period
1101 = Delay by 6.5 Dot-Clock Period
1110 = Delay by 7.0 Dot-Clock Period
1111 = Delay by 7.5 Dot-Clock Period

Figure D-44. Mesh Row Enable Register

Register 07h: Bank F (MC_Reg)
Mesh Control Register (Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

D0: Mesh Enable
0 = Mesh is Disabled
1 = Mesh is Enabled

D1: Software Mesh
0 = Hardware Defines Field Number
1 = Software Defines Field Number

D2: Software Field Number/Polarity of Halftone Output
0 = Even Field/Positive Halftone Output
1 = Odd field/Negative Halftone Output

D3: P20 for Halftoning
0 = Normal Mesh Effect
1 = Use P20 Output for Halftoning

D6–D4: Mesh Color

D7: Halftone Output Delay on P20
xx/x = Bits 5, 4 in ROW_SPACE/(7)
00/0 = No Delay
00/1 = Delay by 0.5 Dot-Clock Period
01/0 = Delay by 1.0 Dot-Clock Period
01/1 = Delay by 1.5 Dot-Clock Period
10/0 = Delay by 2.0 Dot-Clock Period
10/1 = Delay by 2.5 Dot-Clock Period
11/0 = Delay by 3.0 Dot-Clock Period
11/1 = Delay by 3.5 Dot-Clock Period

Figure D-45. Mesh Control Register

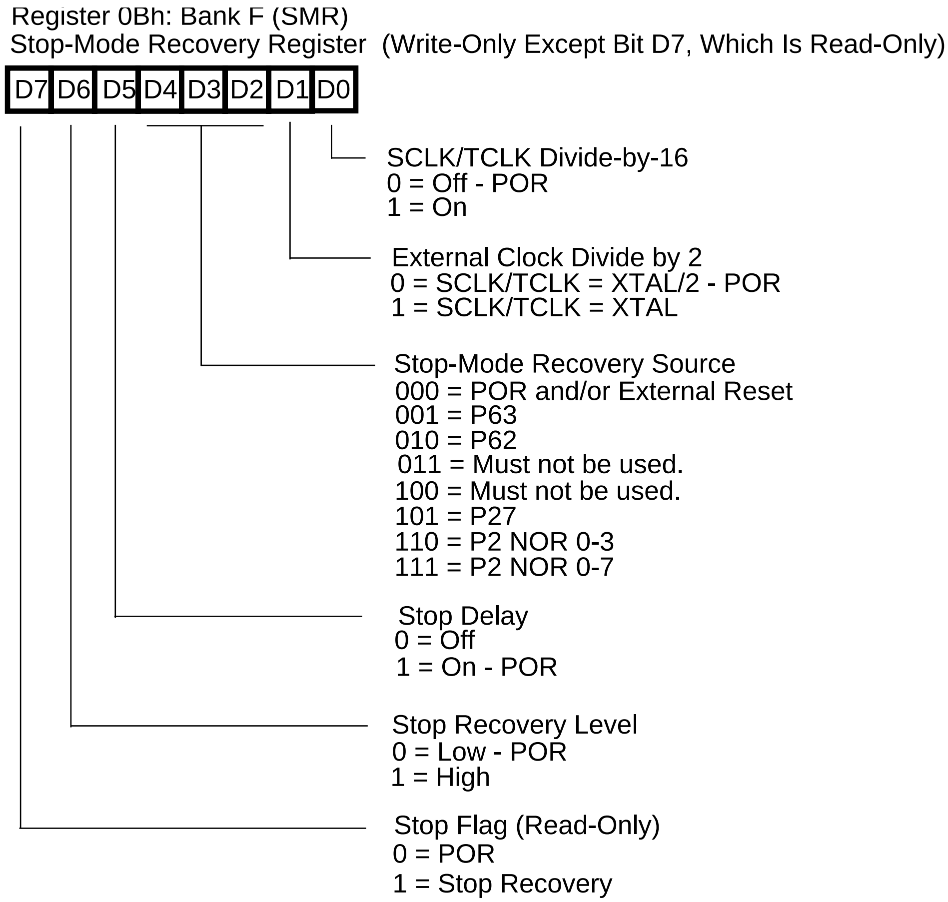


Figure D-46. Stop-Mode Recovery Register

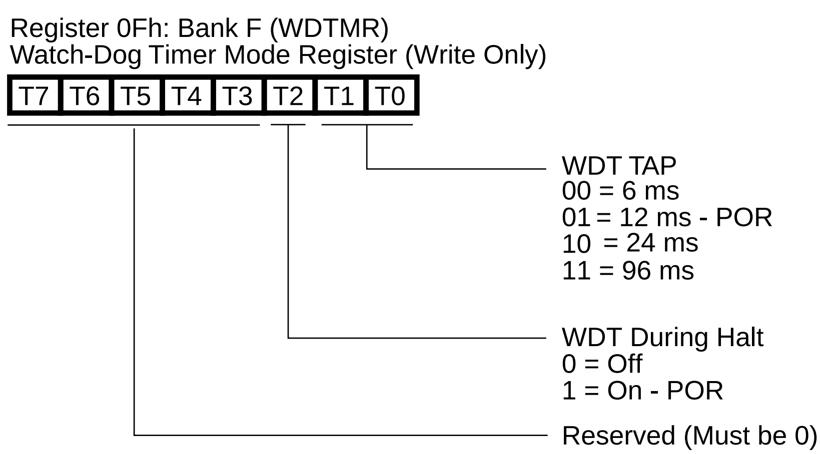


Figure D-47. Watch-Dog Timer Mode Register

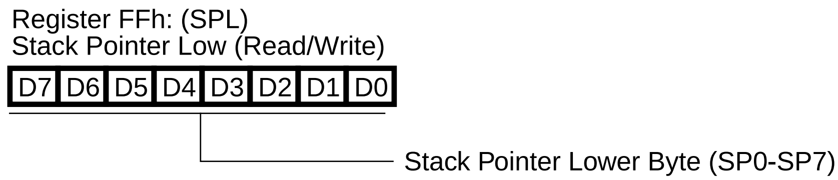


Figure D-48. Stack Pointer Low Register

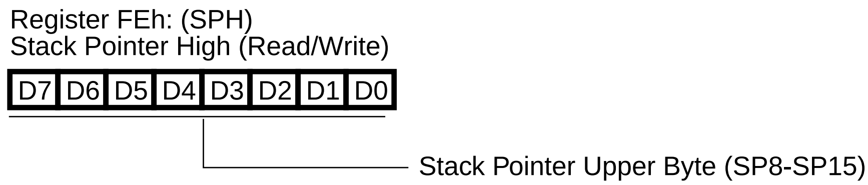


Figure D-49. Stack Pointer High Register

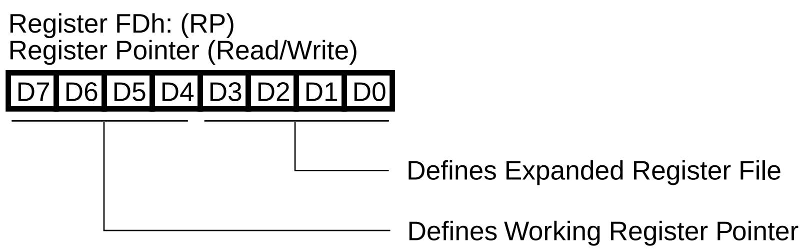


Figure D-50. Register Pointer

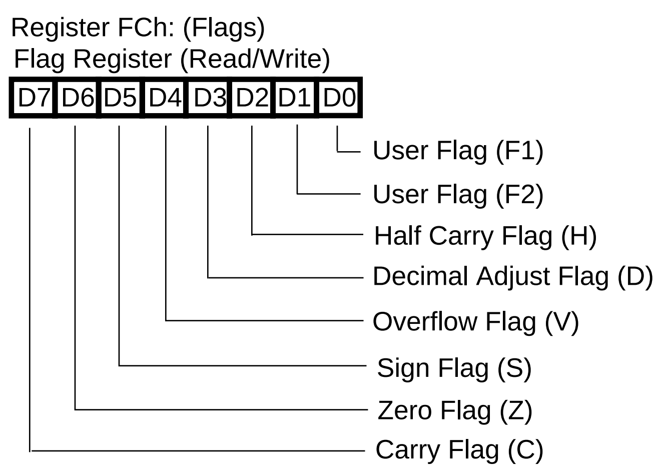


Figure D-51. Flag Register

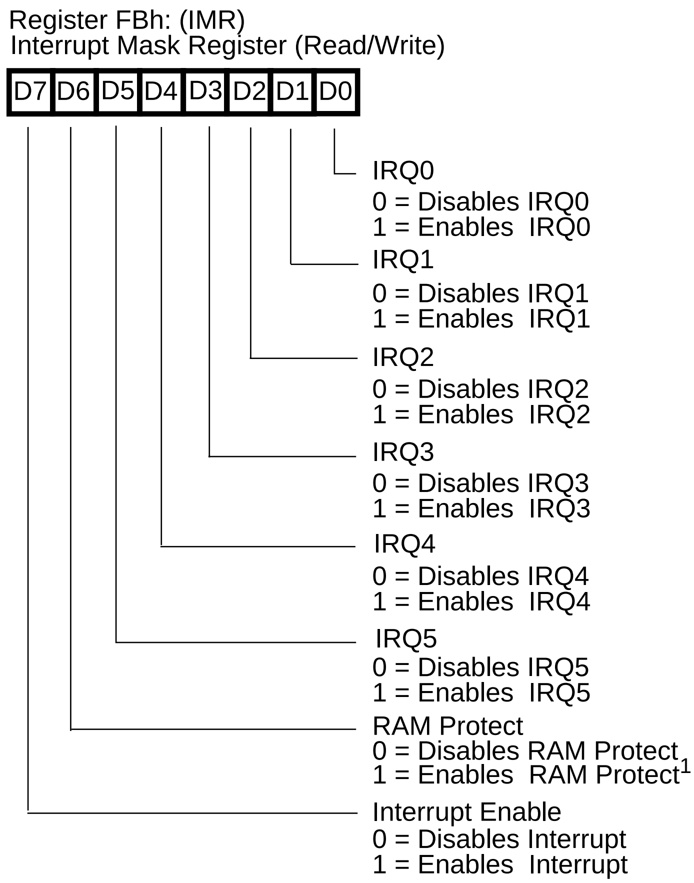


Figure D-52. Interrupt Mask Register

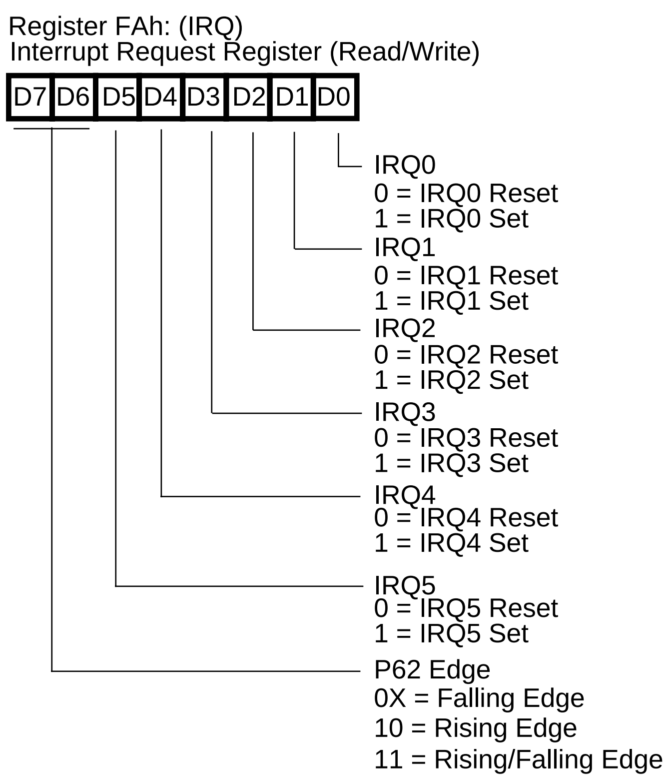


Figure D-53. Interrupt Request Register

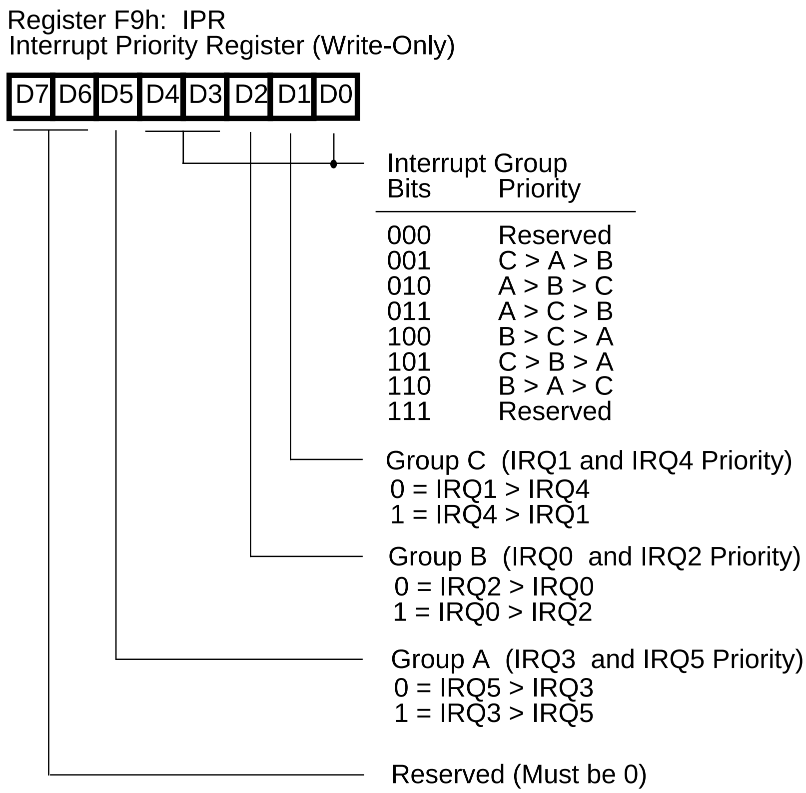


Figure D-54. Interrupt Priority Register

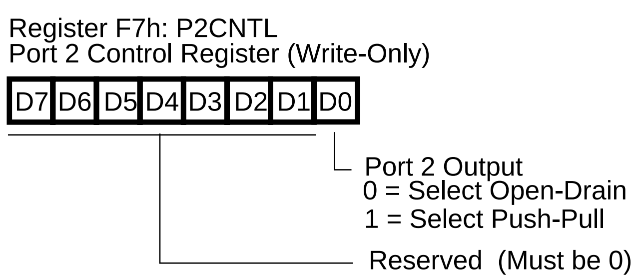


Figure D-55. Port 2 Control Register

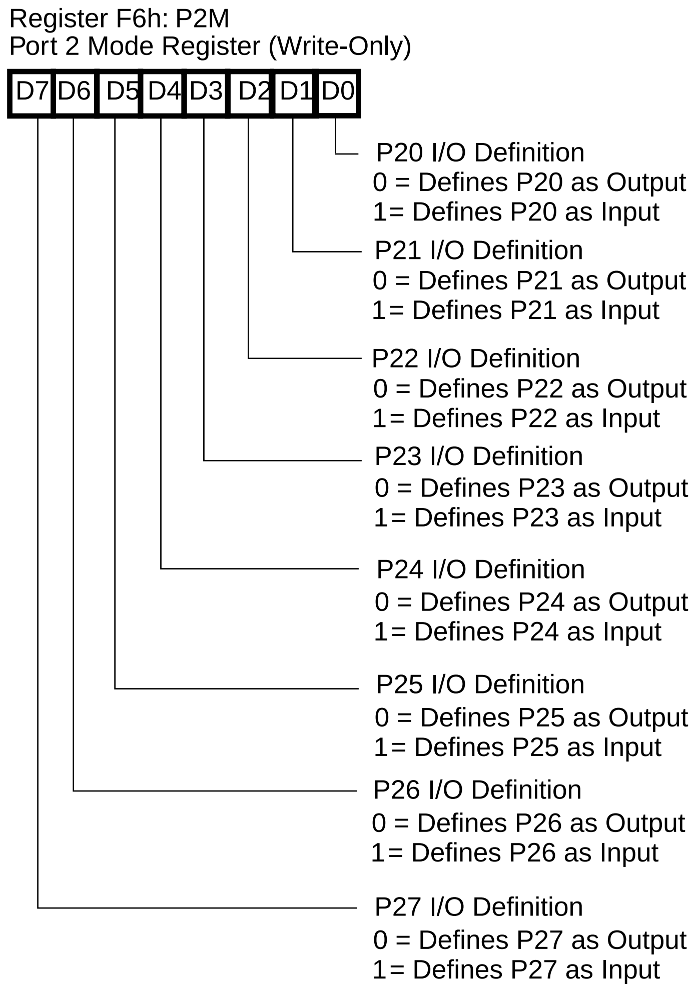


Figure D-56. Port 2 Mode Register

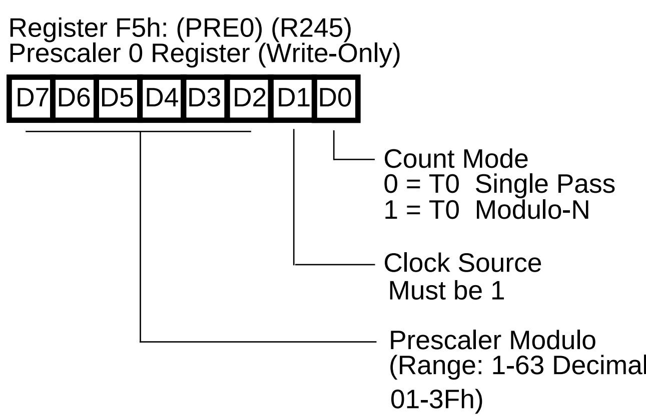


Figure D-57. Prescaler 0 Register

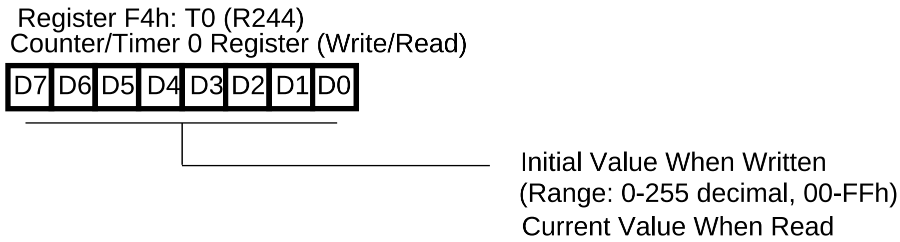


Figure D-58. Counter/Timer 0 Register

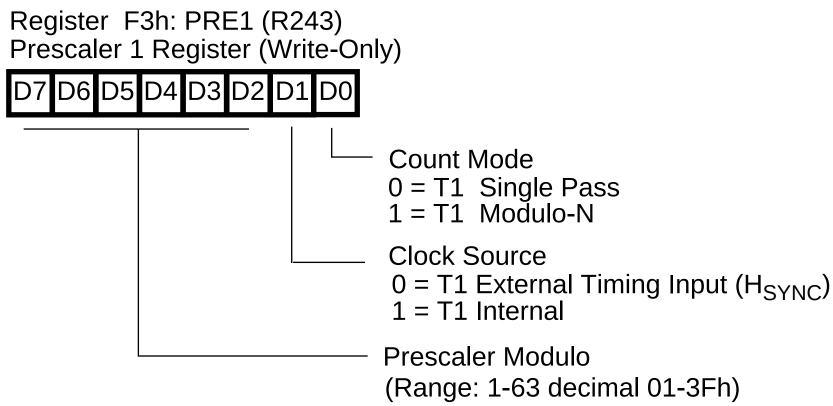


Figure D-59. Prescaler 1 Register

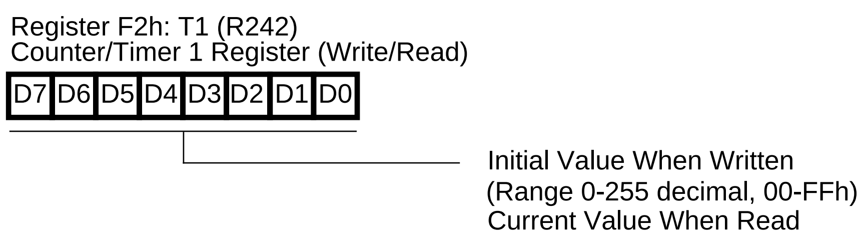


Figure D-60. Counter/Timer 1 Register

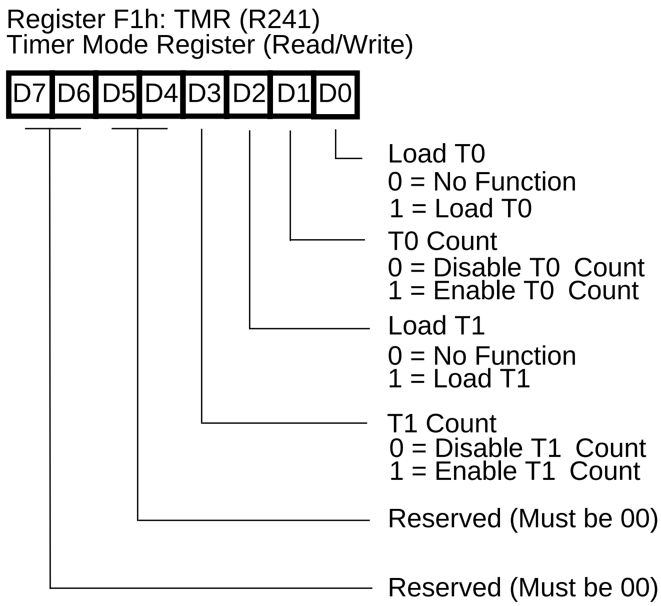


Figure D-61. Timer Mode Register

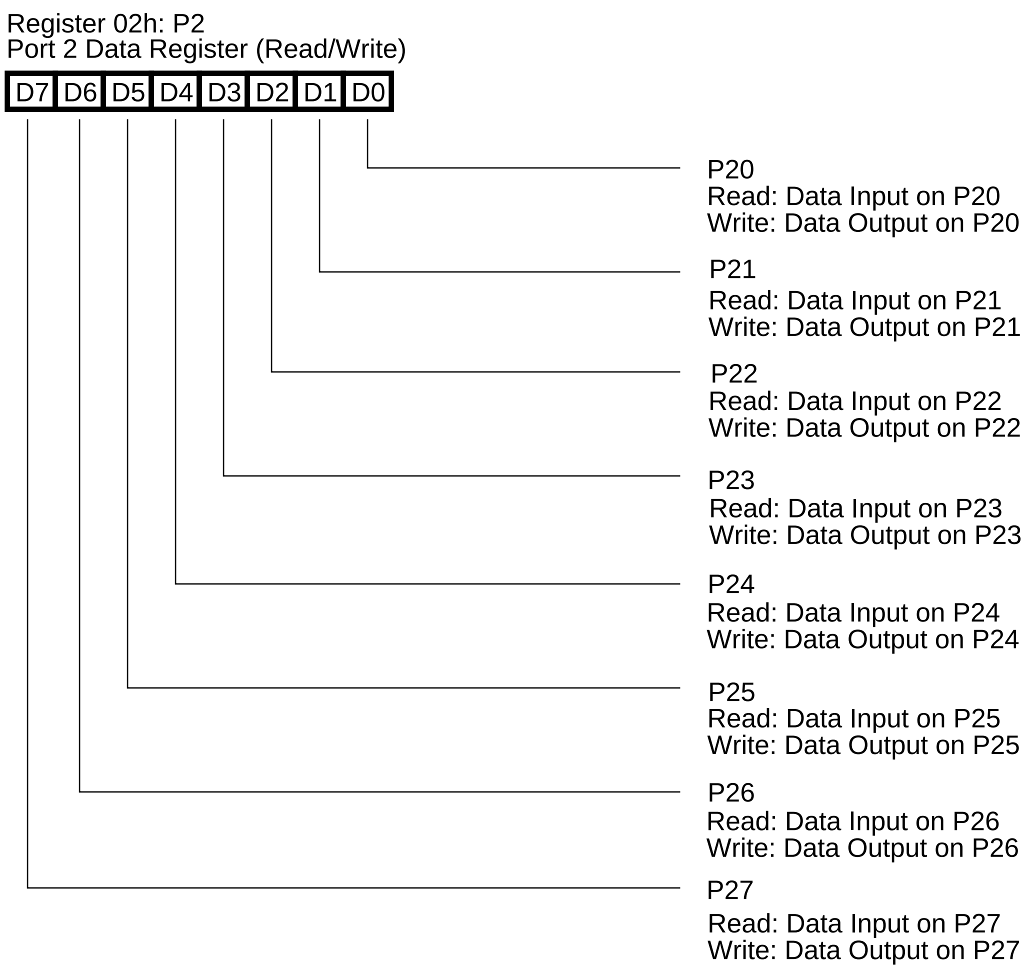


Figure D-62. Port 2 Data Register

APPENDIX E
EMI/NOISE REDUCTION

E.1 EMI/NOISE REDUCTION THROUGH PCB DESIGN

Z90230 family is a complicated mixed signal device. The performance of analog circuitries can be very susceptible to external noise. Digital circuits can generate high frequency noises and EMI from other components in the PCB can deteriorate performance.

For the best EMI performance, PCB design needs to be done for sufficient decoupling of noise from the microcontroller. That noise can be picked up by external circuitry if it does not have good decoupling.

High EMI immunity means good resistiveness that results from:

- Decoupling
- PCB Layout

The following figure demonstrates an effective decoupling scheme:

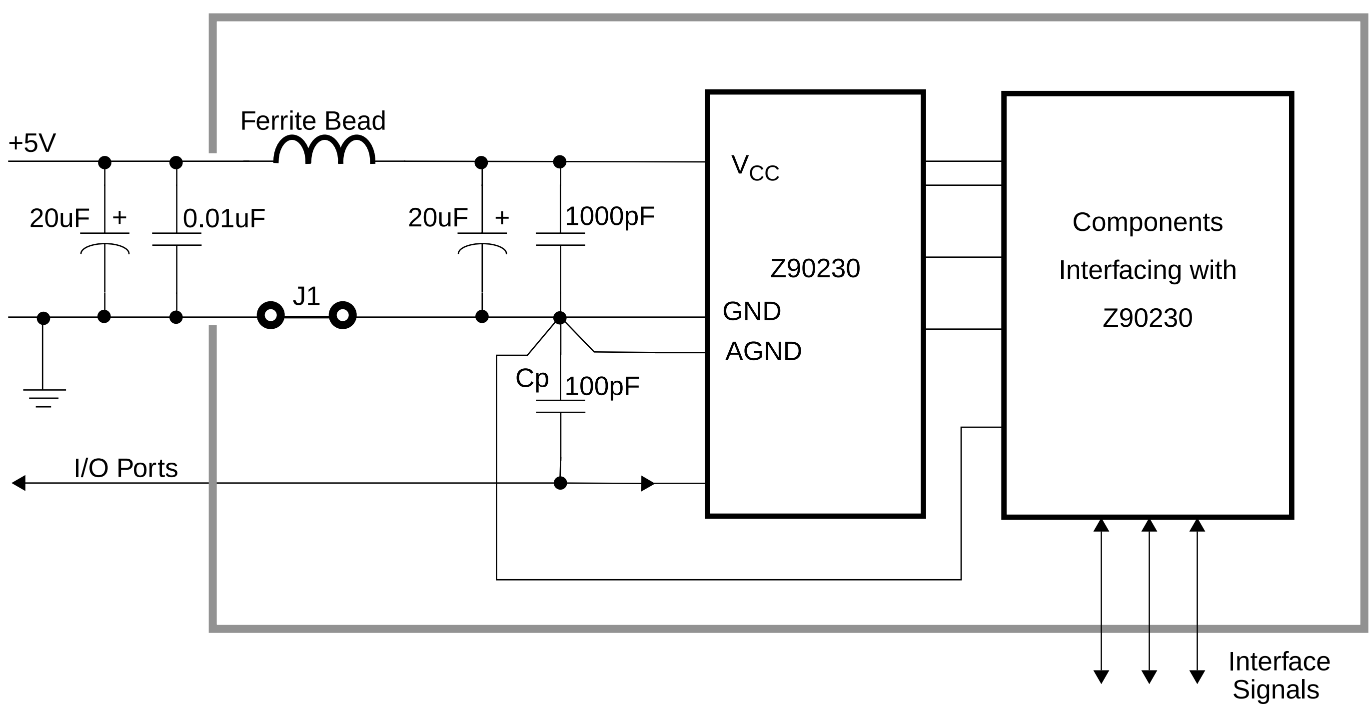


Figure E-1. Application Circuit

Figure E-1 illustrates a sample PCB layout. Only one power line is needed to minimize interference from other circuits. J1 is a virtual shunt on the ground line for PCB auto layout to group grounds.

With this design, the device requires only one common ground, which is connected to the chassis ground.

Port outputs can be very susceptible to high frequency noise, and can radiate EMI to the long path and capacitive loading termination. Wires close to the ports could induce interference. The design should include surface mount capacitors (Cp) on the ports side of the board, with a short path between the ground and the ports.

Power decoupling should be done with ferrite beads, electrolyte capacitors, and ceramic capacitors.

Z90230 Family of DTCs
User's Manual

Index

A

AC Characteristics 2-28
Address Space 3-1
Addressing Modes
  16-Bit Register 3-2
  4-Bit Indirect 3-48
  8-Bit Indirect Register 3-47
  8-Bit Register 3-45
  Direct 3-50
  Immediate Data 3-52
  Indexed 3-49
  Indirect 3-46
  Introduction 3-45
  Relative 3-51
  Working Registers 3-45
Analog-to-Digital Converter
  Block Diagram B-3
  Data Register 2-11, B-2
  Reference B-1
Assembly Language Syntax 3-58

B

Bar Display 4-5
Binary encoding 3-57
Block Diagram
  Device 2-10
  Interrupt 3-30

C

CALL
  Direct Addressing Mode 3-50
  Stack 3-11
CGROM 4-19
Character
  Cell Resolution 4-18
  Pixel Map 4-19
Character Size and Smoothing 4-20
Clock
  External 3-15
  HSYNC Input Mode 3-27
  LC 3-15
  Stop-Mode Recovery 3-43
  XTAL from LC Oscillator 3-16
Color Palette Selection Bits 4-24
Condition Codes 3-56
Core Customization 2-9
Counter/Timers
  Block Diagram 3-22
  Continuous Mode 3-26
  Description 3-21
  Operation 3-24
  Register Map 3-23
  Single-Pass Mode 3-26
Crystals and Resonators 3-15

D

Decrement and Jump If Not Zero 3-51
Design
  Circuit Board Rules 3-14
  Indications of Unreliability 3-13
Digital/Analog Conversion via PWM 8-7
Direct Register Map 3-50
Disable
  Interrupt 3-34
  Interrupts and Polled Processing 3-40

E

EMI/Noise Reduction E-1

Enable
- Count 3-26
- Interrupt 3-34
- Interrupt and Polled Processing 3-40
- Interrupts Instruction and IRQ Register 3-35

Error Conditions 3-4
Expanded Register File 2-14, 3-5

F

Fade 4-15
Features 1-1
Flags
- Carry 3-55
- Condition Codes 3-56
- Decimal Adjust 3-55
- Definitions 3-56
- Half Carry 3-55
- Overflow 3-55
- Processor 3-54
- Register 3-54
- Settings Definitions 3-56
- Sign 3-55
- Zero 3-55

Frame
- Data Read Flowchart 5-9
- Data Write Flowchart 5-8

Fringing 4-21

G

General-Purpose Registers 3-3
Global Interrupt Enable 3-35

H

Halftone 4-7
Halt-Mode Operation 3-41

I

I^2C
- Concepts 5-1
- Data Transfer 5-1
- Master Interface 5-3
- Software Control 5-6

ICE Chip 2-4
Infrared
- Decoding 7-5
- Interface 7-1
- Remote Control Interface 7-1

Initialization Code for IRQ Register 3-36
Instructions
- Arithmetic 3-53
- Bit Manipulation 3-53
- Block Transfer 3-53
- CPU Control 3-54
- Functional Summary of Instruction Set 3-52
- Interrupt Return 3-34
- Load 3-53
- Logical 3-53
- Program Control 3-53
- Rotate and Shift 3-53
- Test Under Mask 3-40

Interrupts
- Acknowledge Timing 3-40
- Control registers 3-30
- Description 3-30
- Effects on Stack 3-38
- Initialization 3-32
- IRQ0-IRQ2 Block Diagram 3-31
- Mask Register Initialization 3-34
- Polled Processing 3-40
- Priority Register Initialization 3-33
- Processing 4-27
- Request Register Configuration 2-23, 3-37
- Request Register Initialization 3-35
- Request Register Logic and Timing .. 3-32
- Request Register Map 2-22, 3-36
- Request Register Reset Functional Logic Diagram 3-37
- Return Instruction 3-34
- Service Routine 7-3
- Software Generation 3-37
- Sources 3-30
- Vectored Cycle Timing 3-39
- Vectored Nesting 3-40

J

Jump
- Conditional and Direct Addressing 3-50
- Conditional and Flag Register 3-54
- Relative 3-51

L

LC Network/Oscillator 3-16

M

Mask
    Operation 3-2
    Programmable ROM 3-10
    Test Under Instruction 3-40
Mesh Effect 4-7

N

Nesting Vectored Interrupts 3-40
Notation 3-57

O

One-Time Programmable ROM 3-10
On-Screen Display Format 4-2
Oscillator
    Control 3-12
    Crystal/Ceramic Resonator 3-15
    Layout 3-13
    LC 3-16
    Operation 3-12

P

Pierce Oscillator with Internal Feedback
    Circuit 3-13
Pin
    124-Pin Pin-Out Diagram 2-4
    Assignment 2-4
    Descriptions 2-7
    Z90231 and Z90233 Identification 2-2
Pointer
    Full Register File 3-4
    Stack 3-11
    Working Register Group 3-3
Polled Processing 3-40
Port
    4, Pin-Out Selection Register 2-13, 8-2
    Configuration Register 3-12
    I/O Ports 6-1
    Pin-Out Selection Register 6-3
Power-On Reset
    Circuit 3-21
    Reset Pin, WDT, and Stop-Mode
        Recovery 3-16
Prescaler
    And Counter/Timers 3-22
    Operations 3-25
    Register and Timer Input 3-27
Product Summary 1-3
Program Memory
    Interrupt Vectors 3-10
    Map 3-10
Pulse Width Modulators 8-1
PUSH 3-11
PWMs
    Mode Register 8-1
    PWM1 through PWM11 8-3
    Settings in the PIN_SLT Register D-16
    Timing Diagram (14 Bit) 8-5
    Timing Diagram (6 Bit) 8-4

R

RAM Protect
    Enable 3-3
    Interrupt Mask Register 3-35
Register
    ADC Data 2-11, 2-12, B-2
    Addressing 3-45
    Bank A 3-8
    Bank B 3-9, 8-1
    Bank C 3-9
    Bar Control 4-5
    Bar Position 4-6
    Color Index 4-25
    Counting Mode 3-25
    Display Attribute 4-22
    Expanded File 2-15
    Expanded File Map 2-14, 3-6
    Expanded Z8 3-8
    Fade Position 1 4-17
    Fade Position 2 4-17
    Flag 3-54
    Horizontal Position 4-4
    Interrupt Control 3-30
    Interrupt Mask Initialization 3-34
    Interrupt Priority Initialization 3-33
    Interrupt Request 3-35
    Interrupt Request Configuration   2-23, 3-37

Register (Continued)
- Interrupt Request Logic and Timing .. 3-32
- Interrupt Request Map 2-22, 3-36
- Interrupt Status 4-28
- IR Capture 0 7-4
- IR Capture 1 7-4
- Master I²C Control 5-5
- Mesh Column End 4-10
- Mesh Column Start 4-9
- Mesh Control 4-11
- Mesh Row Enable 4-10
- Port 4 Data 6-5
- Port 4 Direction Control 6-6, 6-9
- Port 4 Pin-Out Selection 2-13, 6-3, 8-2
- Port Configuration 6-1
- Prescaler 1 3-27
- PWM Mode 8-2
- PWM1 through PWM10 8-6
- PWM11 8-7
- Quick Reference D-1
- Row Attribute 4-27
- Standard Z8 3-8
- Stop-Mode Recovery 2-16, 3-21, 3-43
- Timer Control 0 7-2
- Timer Control 1 7-3
- Timer Mode 3-25
- Timer Mode Reset 3-29
- Vertical Position 4-4

Reset
- Circuit 3-18
- Interrupt Conditions 3-41
- IRQ Functional Logic Diagram 3-37
- IRQ Register 3-35
- Pin, Internal POR 3-17
- Power-On Timer 3-21
- Prescaler 0 3-29
- Prescaler 1 3-28
- Timer Mode Register 3-29
- Timing 3-18
- Values ERF Bank 0 3-19

Resonators 3-15
RETURN 3-11

S

Second Color 4-5
Smoothing and Character Size 4-20
Stack
- Effects of Interrupt 3-38
- Operations 3-11

Stop-Mode
- Operation 3-41
- Recovery Circuit 3-42
- Recovery Prescaler 3-26
- Recovery Register 3-42
- Recovery Register and POR 3-21
- Recovery Source 3-43

Symbols 3-58
Synchronization Specification 4-29
Syntax of Assembly Language 3-58

V

Vectored Processing 3-38
VRAM
- Address Map 4-23
- Mode 4-23

W

Watch-Dog Timer
- Description 3-20
- During HALT 3-20
- Instruction and Flags 3-55
- Mode Register 3-20
- Time-Out Period 3-20

Working Register Groups 3-3

Zilog, Inc. 210 East Hacienda Ave.
Campbell, CA 95008-6600
Telephone (408) 370-8000
FAX (408) 370-8056
Internet: http://www.zilog.com