CEG 3136 – Computer Architecture II
Tutorial 3 – The Parallel Ports
Fall 2019

Figure 1-1 MC9S12DT256 Block Diagram

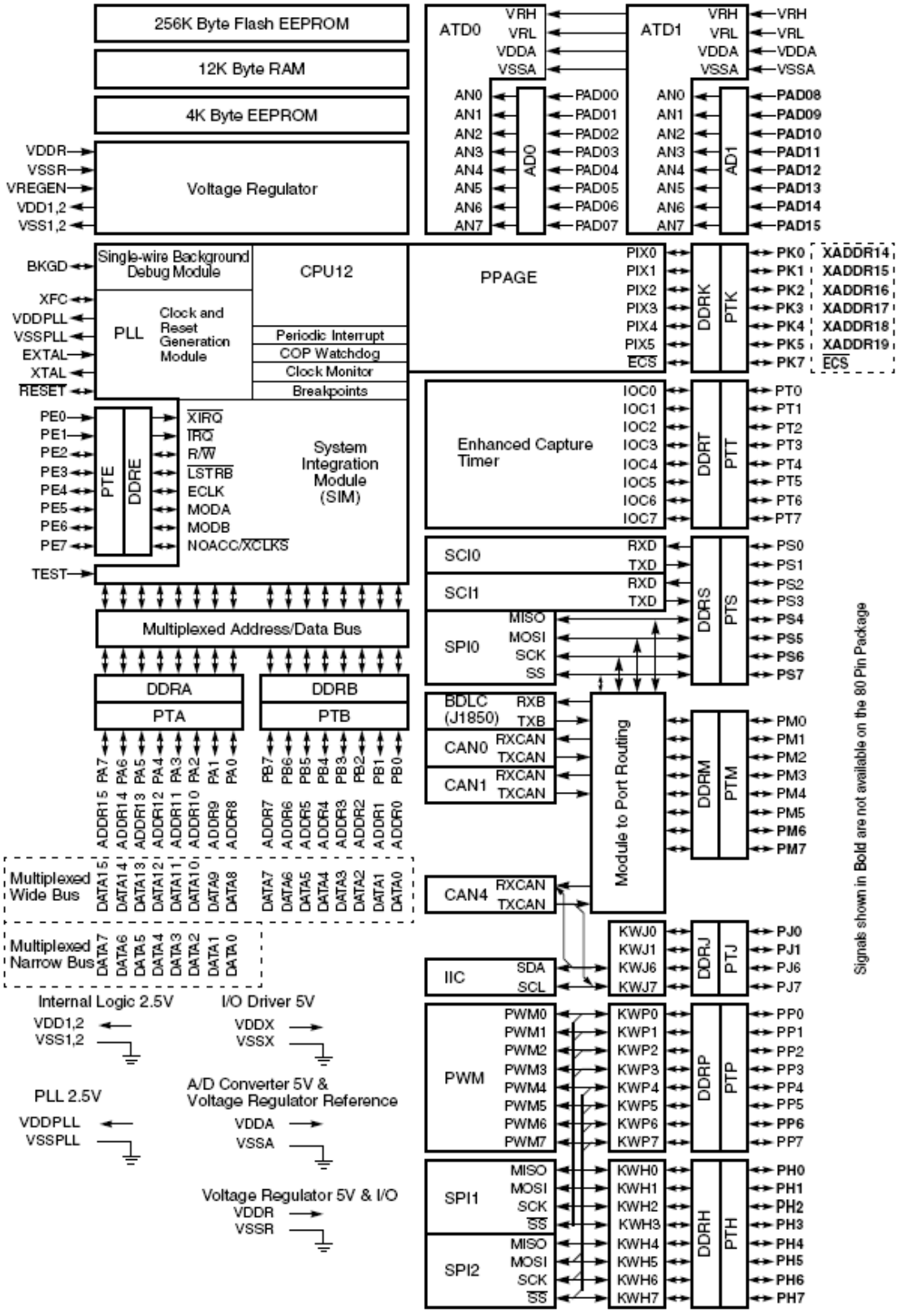


Table 0-1 shows the availability of peripheral modules on the various derivatives. For details about the compatibility within the MC9S12D-Family refer also to engineering bulletin EB386.

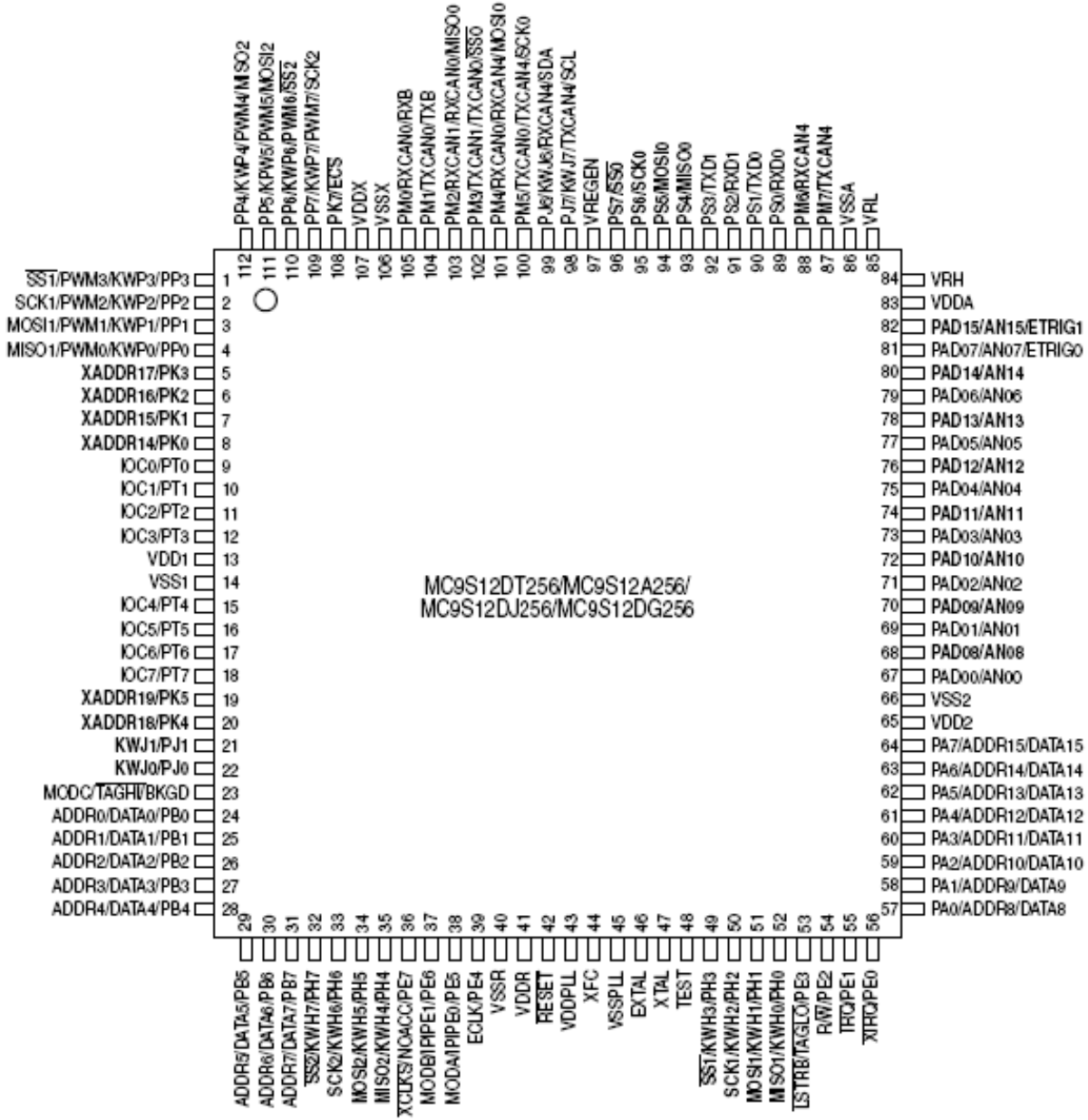
Table 0-1 Derivative Differences

Generic device	MC9S12A256	MC9S12DT256	MC9S12DJ256	MC9S12DG256
# of CANs	0	3	2	2
CAN0	—	✓	✓	✓
CAN1	—	✓	—	—
CAN4	—	✓	✓	✓
J1850/BDLC	—	—	✓	—
Package	112 LQFP/80 QFP	112 LQFP/80 QFP	112 LQFP/80 QFP	112 LQFP/80 QFP
Mask set	L91N/L01Y	L91N/L01Y	L91N/L01Y	L91N/L01Y
Temp Options	C	M, V, C	M, V, C	M, V, C
Package Code	PV/FU	PV/FU	PV/FU	PV/FU
Notes	An errata exists contact Sales Of ce	An errata exists contact Sales Of ce	An errata exists contact Sales Of ce	An errata exists contact Sales Of ce

The following figure provides an ordering number example for the MC9S12H-Family devices.

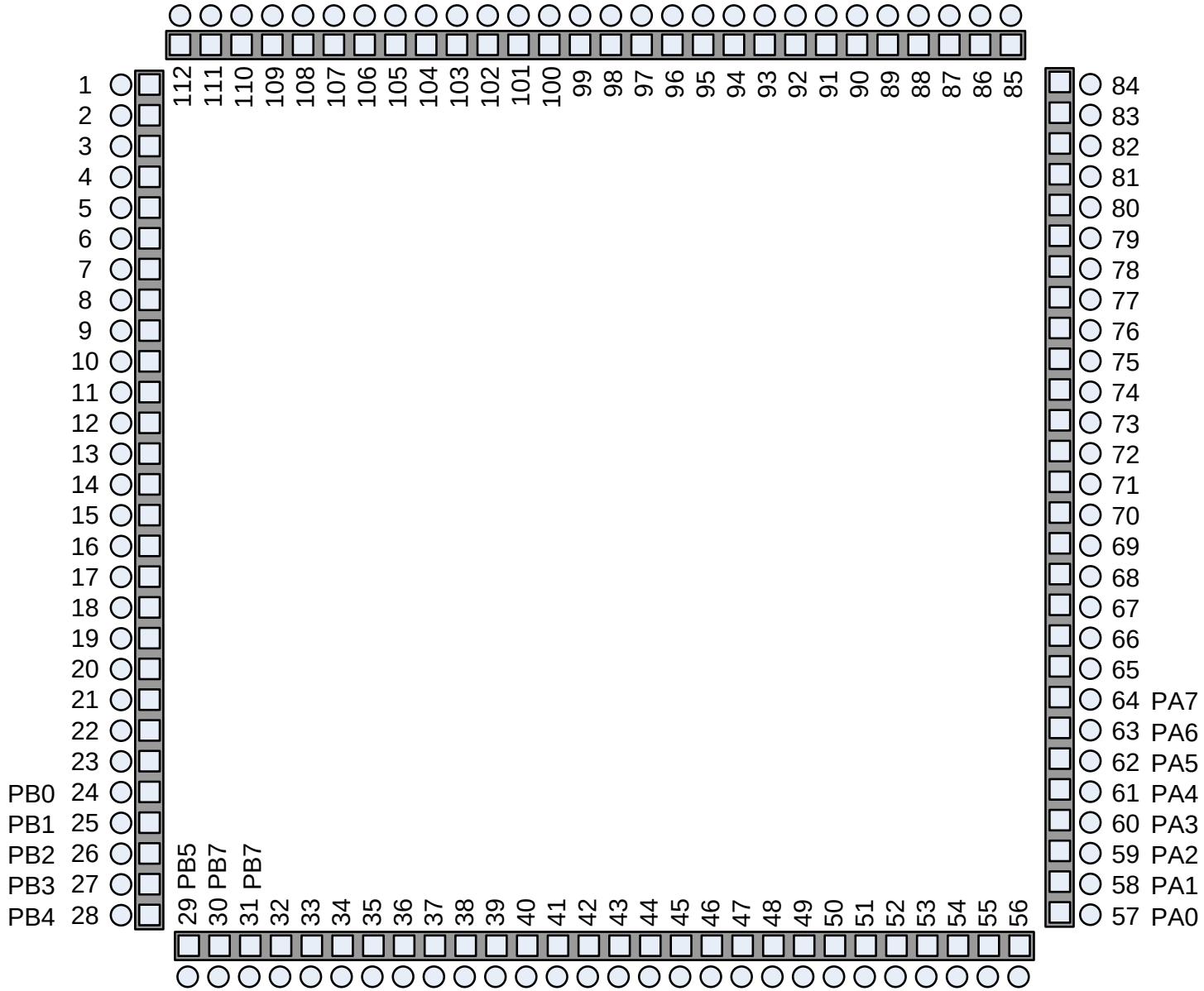
Figure 0-1 Order Partnumber Example

Pin Assignments in 112 pin package



Dragon-12 Header Blocks (male/female)

The pins correspond to the layout of the microcontroller shown on the previous page,.



Chapter 2
Register Block

2.1 Introduction

The register block can be mapped to any 2-Kbyte boundary within the standard 64-Kbyte address space by manipulating bits REG15–REG11 in the register initialization register (INITRG). INITRG establishes the upper five bits of the register block's 16-bit address. The register block occupies the first 512 bytes of the 2-Kbyte block.

Default addressing (after reset) is indicated in Figure 2-1. For additional information, refer to Chapter 5 Operating Modes and Resource Mapping.

NOTE

In expanded and peripheral modes, these registers are not in the map:

- Port A data register, PORTA
- Port B data register, PORTB
- Port A data direction register, DDRA
- Port B data direction register, DDRB

In peripheral mode or in expanded modes *with the emulate port E bit (EME)* set, these registers are not in the map:

- Port E data register, PORTE
- Port E data direction register, DDRE

In peripheral mode, these registers are not in the map:

- Mode register, MODE
- Pullup control register, PUCR
- Reduced drive register, RDRIV

Section 3 Memory Map/Register Definition

A summary of the registers associated with the MEBI sub-block is shown in **Figure 3-1**. Detailed descriptions of the registers and bits are given in the subsections that follow.On most chips the registers are mappable. Therefore, the upper bits may not be all zeros as shown in the table and descriptions.

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
$0000	PORTA	Read Write	Bit 7	6	5	4	3	2	1	Bit 0
$0001	PORTB	Read Write	Bit 7	6	5	4	3	2	1	Bit 0
$0002	DDRA	Read Write	Bit 7	6	5	4	3	2	1	Bit 0
$0003	DDRB	Read Write	Bit 7	6	5	4	3	2	1	Bit 0
$000C	PUCR	Read Write	PUPKE	0	0	PUPEE	0	0	PUPBE	PUPAE

3.1.1 Port A Data Register (PORTA)

Address: Base + $__00

	BIT 7	6	5	4	3	2	1	BIT 0
Read: Write:	Bit 7	6	5	4	3	2	1	Bit 0
Reset:	—	—	—	—	—	—	—	—
Single Chip:	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
Expanded Wide, Emulation Narrow with IVIS, and Peripheral:	AB/DB15	AB/DB14	AB/DB13	AB/DB12	AB/DB11	AB/DB10	AB/DB9	AB/DB8
Expanded Narrow:	AB15 and DB15/DB7	AB14 and DB14/DB6	AB13 and DB13/DB5	AB12 and DB12/DB4	AB11 and DB11/DB3	AB10 and DB10/DB2	AB9 and DB9/DB1	AB8 and DB8/DB0

Figure 3-2 Port A Data Register (PORTA)

Read: anytime when register is in the map

Write: anytime when register is in the map

Port A bits 7 through 0 are associated with address lines A15 through A8 respectively and data lines D15/D7 through D8/D0 respectively. When this port is not used for external addresses such as in single-chip mode, these pins can be used as general-purpose I/O. Data Direction Register A (DDRA) determines the primary direction of each pin. DDRA also determines the source of data for a read of PORTA.

This register is not in the on-chip memory map in expanded and special peripheral modes. Therefore, these accesses will be echoed externally.

3.1.3 Data Direction Register A (DDRA)

Address: Base + $__02

	BIT 7	6	5	4	3	2	1	BIT 0
Read: Write:	Bit 7	6	5	4	3	2	1	Bit 0
Reset:	0	0	0	0	0	0	0	0

Figure 3-4 Data Direction Register A (DDRA)

Read: anytime when register is in the map

Write: anytime when register is in the map

This register controls the data direction for Port A. When Port A is operating as a general-purpose I/O port, DDRA determines the primary direction for each Port A pin. A "1" causes the associated port pin to be an output and a "0" causes the associated pin to be a high-impedance input. The value in a DDR bit also affects the source of data for reads of the corresponding PORTA register. If the DDR bit is zero (input) the buffered pin input state is read. If the DDR bit is one (output) the associated port data register bit state is read.

3.1.10 Pull-Up Control Register (PUCR)

Address: Base + $__0C

	BIT 7	6	5	4	3	2	1	BIT 0
Read:	PUPKE	0	0	PUPEE	0	0	PUPBE	PUPAE
Write:								
Reset:(1)	1	0	0	1	0	0	0	0

= Unimplemented

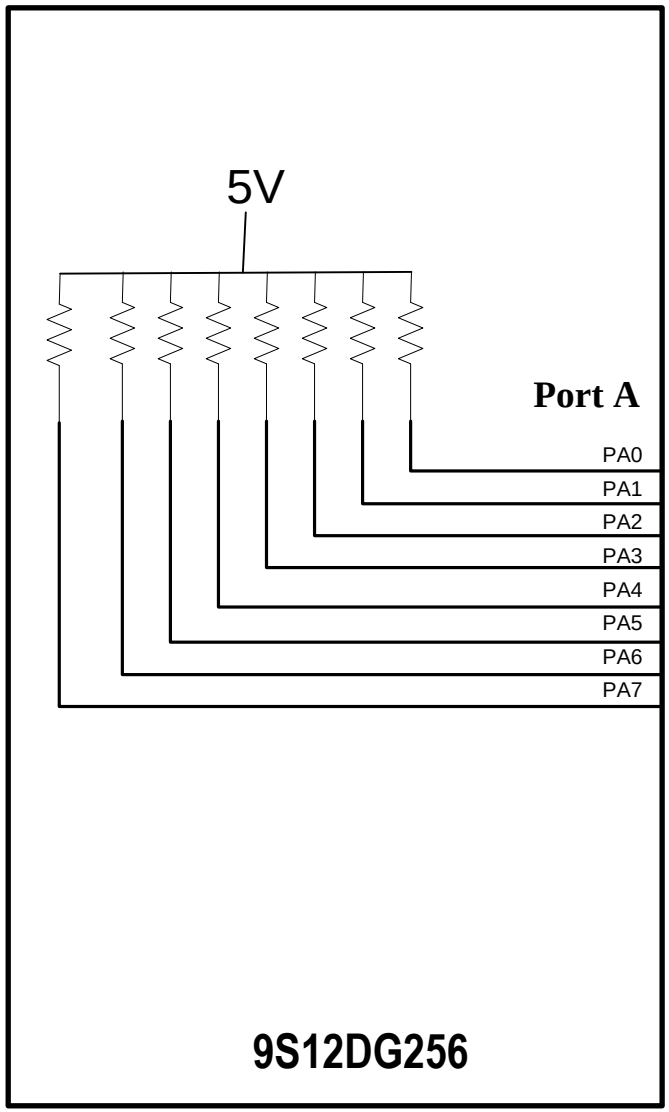
NOTES:
1. The default value of this parameter is shown. Please refer to the specific device User's Guide to determine the actual reset state of this register.

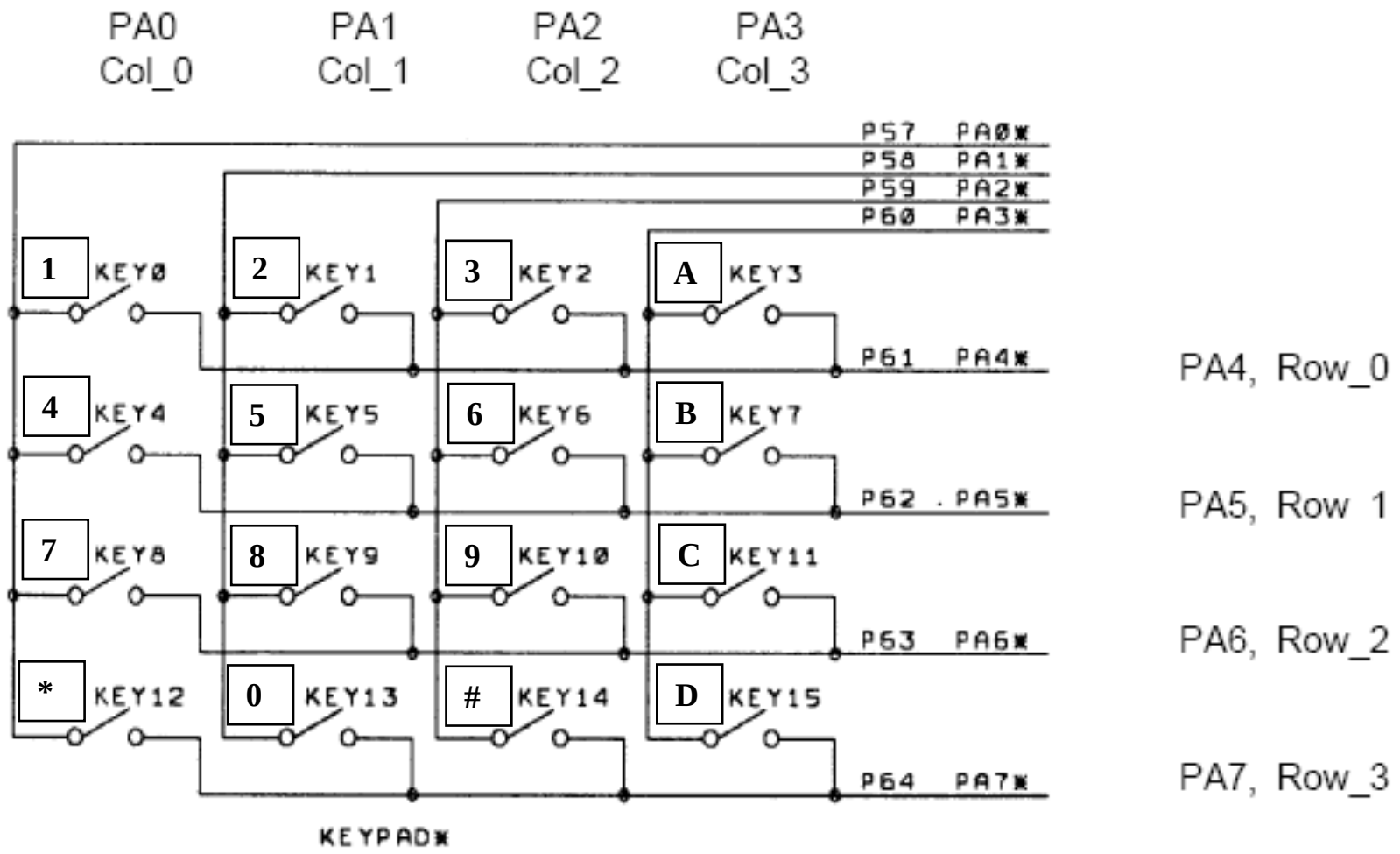
Figure 3-11 Pullup Control Register (PUCR)

Read: anytime (provided this register is in the map).

Write: anytime (provided this register is in the map).

This register is used to select pull resistors for the pins associated with the core ports. Pull resistors are assigned on a per-port basis and apply to any pin in the corresponding port that is currently configured as an input. The polarity of these pull resistors is determined by chip integration. Please refer to the specific device User's Guide to determine the polarity of these resistors.


5V
Port A
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
9S12DG256



Physically, the keypad can be view as eight wires forming a matrix (see the above figure). Each wire is connected to one of the pins. Each key is located over one of the intersections in the matrix. When a key is pressed, the two cooresponding wires are connected and thus so are the corresponding pins.

Questions

Consider loading the values shown into the following registers:
DDRA: Binary value 11110000
PORTA: Binary value 00000000

1) If PUPAE bit in the PUCR register is set to 0, what can be said about the voltage levels that will appear at PA0 to PA3? What if PUPAE is set to 1?
2) What bit patterns would you find in PORTA (data register) when pressing the following keys (assume that PUPAE is set to 1):
    a. 1
    b. 4
    c. 0
    d. C
    e. 3
    f. #
3) Note that some bits patterns are the same for different keys in question 2. Any idea how you could change the PORTA register to be able to differentiate keys?
4) Given the logic from question 3, complete the table on the next page to show what differentiating bit patterns would appear in PORTA when each key is pressed.

Key	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
1								
2								
3								
A								
4								
5								
6								
B								
7								
8								
9								
C								
*								
0								
#								
D								