CP316
Analog to Digital Converters

Terry Sturtevant

Wilfrid Laurier University

February 9, 2018

Analog to Digital Converters

Analog to Digital Converters

overview

Analog to Digital Converters

overview

→ Section 12.3

Analog to Digital Converters

overview

→ Section 12.3

registers

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

integrated sample and hold

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

integrated sample and hold

→ Section 12.3.2

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

integrated sample and hold

→ Section 12.3.2

clocking

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

integrated sample and hold

→ Section 12.3.2

clocking

→ Sections 12.3.3 to 12.3.5

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

integrated sample and hold

→ Section 12.3.2

clocking

→ Sections 12.3.3 to 12.3.5

→ Section 17.0

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

integrated sample and hold

→ Section 12.3.2

clocking

→ Sections 12.3.3 to 12.3.5

→ Section 17.0

10 bit operation

Analog to Digital Converters

overview

→ Section 12.3

registers

→ Section 12.3.1

integrated sample and hold

→ Section 12.3.2

clocking

→ Sections 12.3.3 to 12.3.5

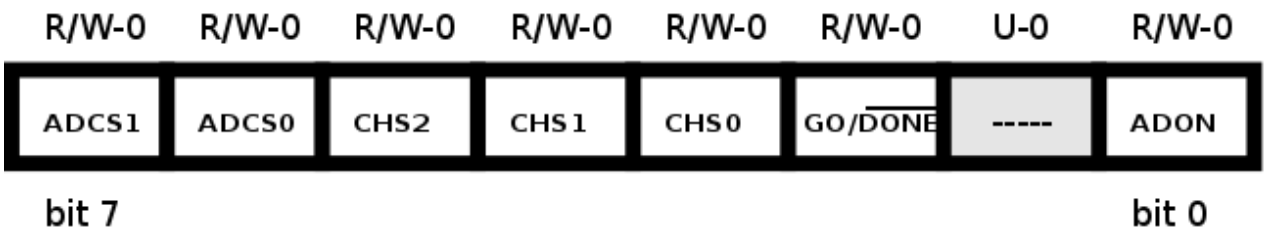
→ Section 17.0

10 bit operation

→ Section 17.4.1

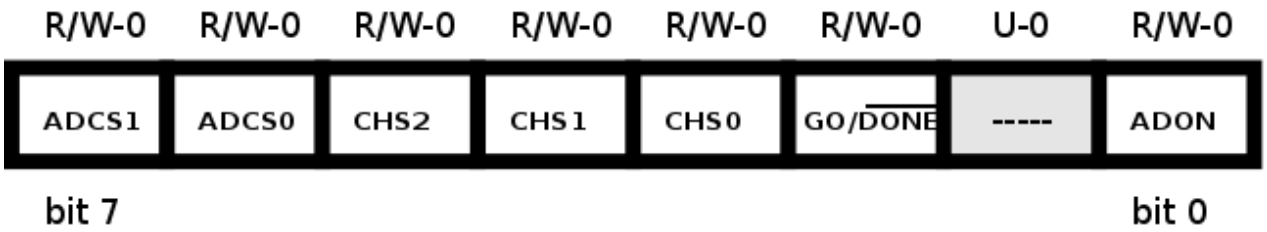
ADCON0

ADCON0


R/W-0
R/W-0
R/W-0
R/W-0
R/W-0
R/W-0
U-0
R/W-0
ADCS1
ADCS0
CHS2
CHS1
CHS0
GO/DONE

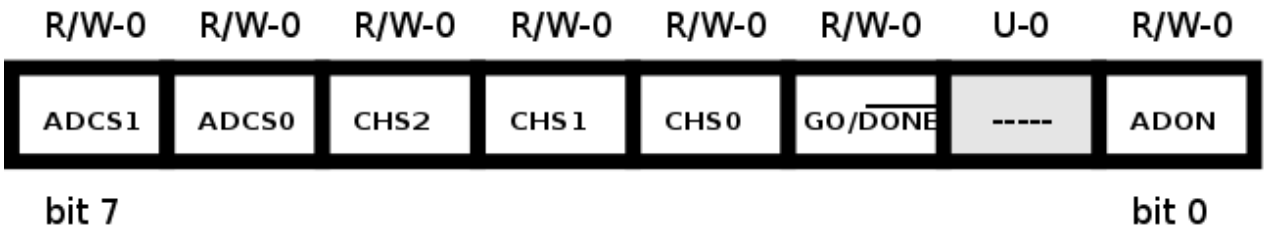
ADON
bit 7
bit 0

ADCON0



Bits in ADCON0 register

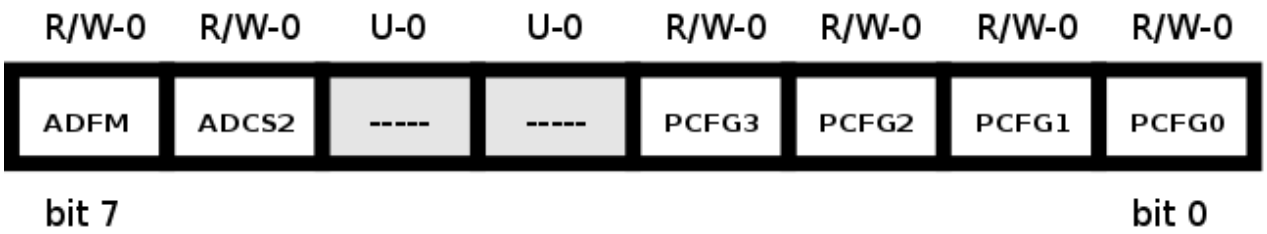
ADCON0



Bits in ADCON0 register - Note ADCS1, ADCS0

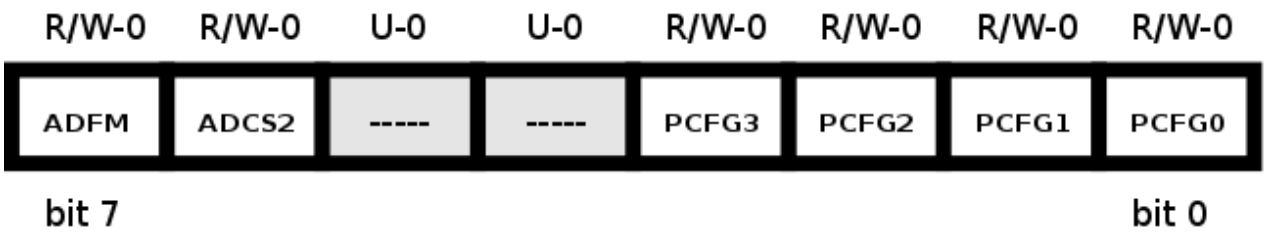
ADCON1

ADCON1


R/W-0
R/W-0
U-0
U-0
R/W-0
R/W-0
R/W-0
R/W-0
ADFM
ADCS2

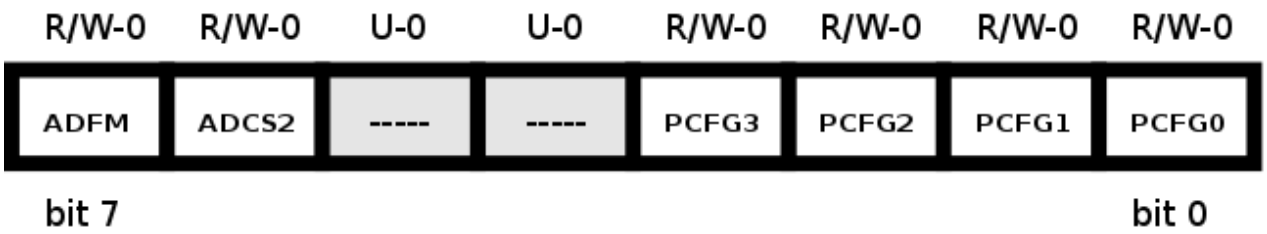
PCFG3
PCFG2
PCFG1
PCFG0
bit 7
bit 0

ADCON1



Bits in ADCON1 register

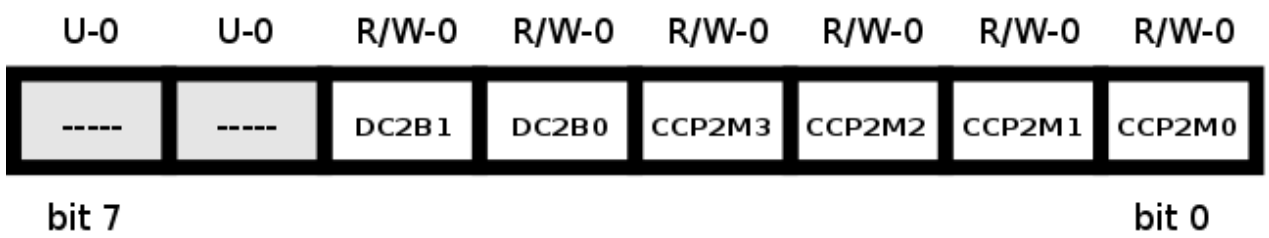
ADCON1



Bits in ADCON1 register - Note ADCS2

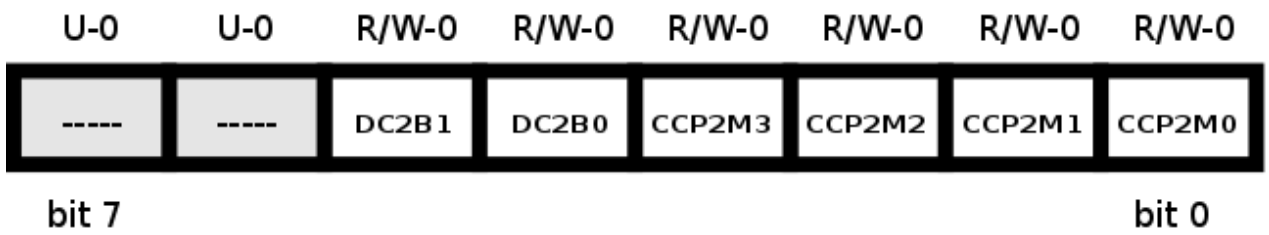
CCP2CON

CCP2CON


U-0
U-0
R/W-0
R/W-0
R/W-0
R/W-0
R/W-0
R/W-0

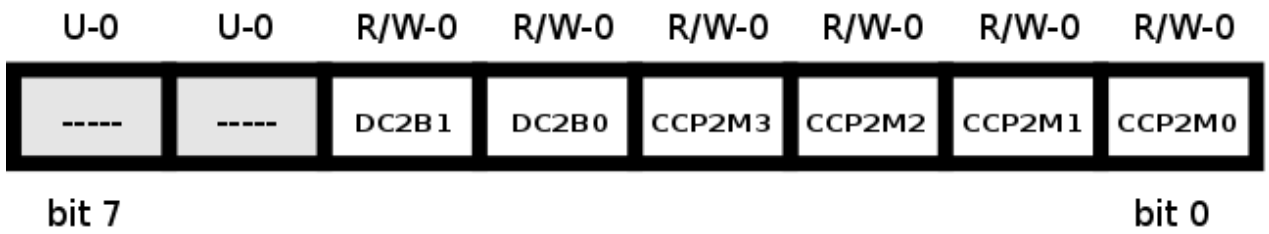
DC2B1
DC2B0
CCP2M3
CCP2M2
CCP2M1
CCP2M0
bit 7
bit 0

CCP2CON



Bits in CCP2CON register

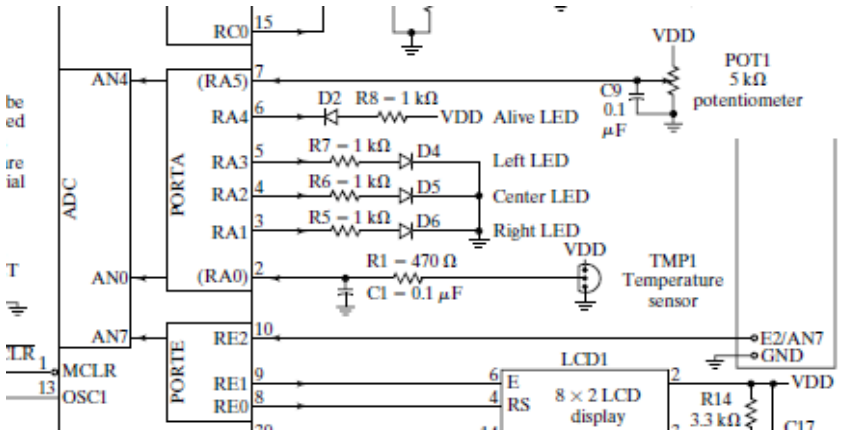
CCP2CON



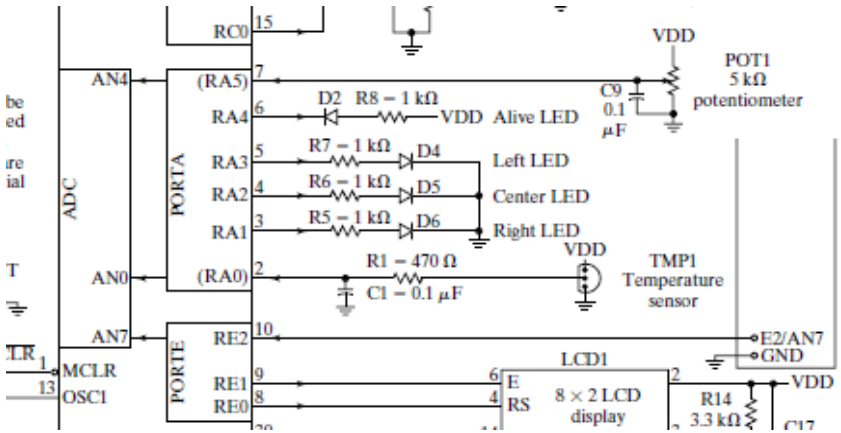
Bits in CCP2CON register -Mode 1011 can start ADC

QwikFlash ADC connections

QwikFlash ADC connections



QwikFlash ADC connections



ramifications???

Initialization

Initialization

overview

Initialization

overview

→ Section 12.4, step 1

Initialization

overview

→ Section 12.4, step 1

interrupts

Initialization

overview

→ Section 12.4, step 1

interrupts

→ Section 6.4.5

Initialization

overview

→ Section 12.4, step 1

interrupts

→ Section 6.4.5

→ Section 17.0

Code

Code

overview

Code

overview

→ Section 12.4

Code

overview

→ Section 12.4

PORT configuration

Code

overview

→ Section 12.4

PORT configuration

→ macro or subroutine?

Code

overview

→ Section 12.4

PORT configuration

→ macro or subroutine?

Initialization

Code

overview

→ Section 12.4

PORT configuration

→ macro or subroutine?

Initialization

→ macro or subroutine?

Code

overview

→ Section 12.4

PORT configuration

→ macro or subroutine?

Initialization

→ macro or subroutine?

Start conversion

Code

overview

→ Section 12.4

PORT configuration

→ macro or subroutine?

Initialization

→ macro or subroutine?

Start conversion

→ macro or subroutine?

Read value

Code

overview

→ Section 12.4

PORT configuration

→ macro or subroutine?

Initialization

→ macro or subroutine?

Start conversion

→ macro or subroutine?

Read value

→ macro or subroutine?

Initialization

Initialization

1. configure ports (**TRISA**, **TRISE**)

Initialization

1. configure ports (**TRISA**, **TRISE**)

   don't inadvertently alter other bits

Initialization

1. configure ports (**TRISA**, **TRISE**)

   don't inadvertently alter other bits
2. set parameters (**ADCON0** , **ADCON1**)

Initialization

1. configure ports (**TRISA**, **TRISE**)

   don't inadvertently alter other bits
2. set parameters (**ADCON0** , **ADCON1**)

   clock source, justification, port configuration

Initialization

1. configure ports (**TRISA**, **TRISE**)

   don't inadvertently alter other bits
2. set parameters (**ADCON0** , **ADCON1**)

   clock source, justification, port configuration
3. set up special event trigger (if desired) (**CCP2CON**)

Initialization

1. configure ports (**TRISA**, **TRISE**)

   don't inadvertently alter other bits
2. set parameters (**ADCON0** , **ADCON1**)

   clock source, justification, port configuration
3. set up special event trigger (if desired) (**CCP2CON**)

   and other related registers

Initialization

1. configure ports (**TRISA**, **TRISE**)

   don't inadvertently alter other bits
2. set parameters (**ADCON0** , **ADCON1**)

   clock source, justification, port configuration
3. set up special event trigger (if desired) (**CCP2CON**)

   and other related registers
4. enable interrupt (if desired) (**PIE1** , **IPR1**)

Initialization

1. configure ports (**TRISA**, **TRISE**)
   don't inadvertently alter other bits
2. set parameters (**ADCON0** , **ADCON1**)
   clock source, justification, port configuration
3. set up special event trigger (if desired) (**CCP2CON**)
   and other related registers
4. enable interrupt (if desired) (**PIE1** , **IPR1**)
   assuming interrupts are enabled globally

Initialization

1. configure ports (**TRISA**, **TRISE**)
   don't inadvertently alter other bits
2. set parameters (**ADCON0** , **ADCON1**)
   clock source, justification, port configuration
3. set up special event trigger (if desired) (**CCP2CON**)
   and other related registers
4. enable interrupt (if desired) (**PIE1** , **IPR1**)
   assuming interrupts are enabled globally
5. choose channel, and start conversion (if desired) (**ADCON0**)

Reading

Reading

Reading is simple

Reading

Reading is simple

1. check flag to see that conversion is done (**PIR1**)

Reading

Reading is simple

1. check flag to see that conversion is done (**PIR1**)
2. get value and process (remember justification) (**ADRESH**, **ADRESL**)

Reading

Reading is simple

1. check flag to see that conversion is done (**PIR1**)
2. get value and process (remember justification) (**ADRESH**, **ADRESL**)
3. clear flag (**PIR1**)

Reading

Reading is simple

1. check flag to see that conversion is done (**PIR1**)
2. get value and process (remember justification) (**ADRESH**, **ADRESL**)
3. clear flag (**PIR1**)
4. change channel (if desired), and start another conversion (if desired) (**ADCON0**)