RTS6: Conception et programmation de Systèmes Embarqués cours 3: périphériques et radio CC2500

Antoine Fraboulet, Fabrice Jumel, Lionel Morel, Tanguy Risset antoine.fraboulet@insa-lyon.fr

Lab CITI, INSA de Lyon



- p. 1/48



TI eZ430-RF2500

TI eZ430-RF2500

TI eZ430-RF2500

- Schématique du module
- PrésentationSchémas

Schéma FET

Brochage et E/S

Exemple

Gestion du temps

Interruptions

Timers

Introduction

Ports série

E/S analogiques





Schématique du module

TI eZ430-RF2500

● TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

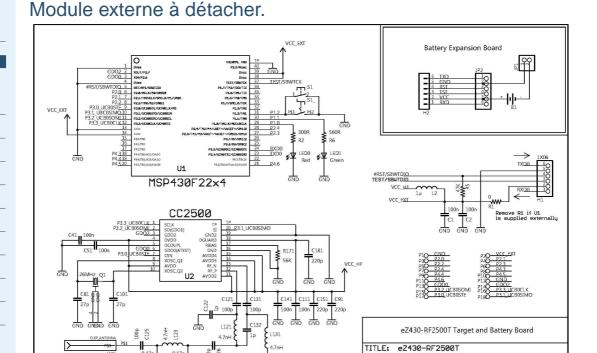
Gestion du temps

Interruptions

Ports série

E/S analogiques

Pilotes de périphériques

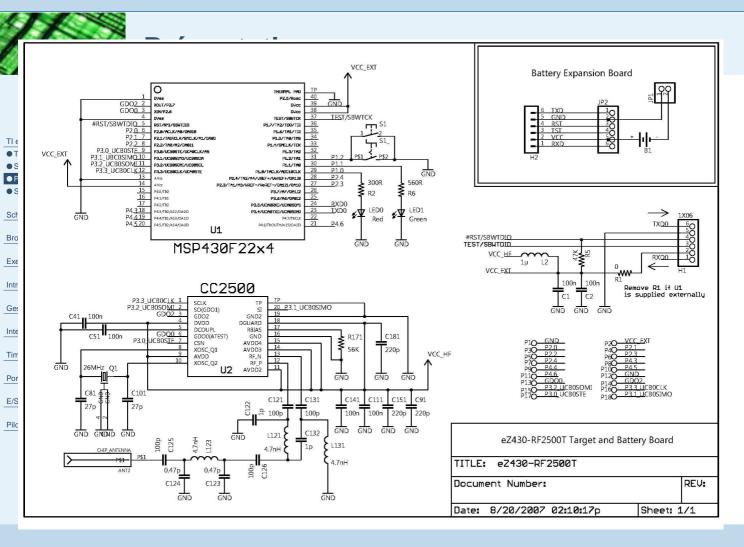


Document Number:

Date: 8/20/2007 02:10:17p

- p. 3/48

REV:





Schémas

TI eZ430-RF2500

- TI eZ430-RF2500
- Schématique du module
- Présentation

Schéma FET

Brochage et E/S

Exemple

Introduction

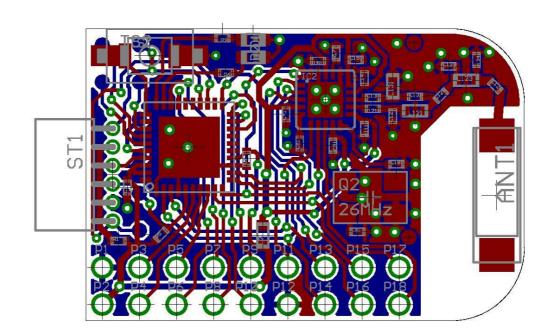
Gestion du temps

Interruptions

Ports série

E/S analogiques

Pilotes de périphériques



- p. 5/48



Schémas partie USB (1)

01 10n GND R2 R3 C4 AVEC AVES 1u/6.3U U1 Gestion du temps MSP430F16× GND3 GND3 GND2 GND1 SCL SDA eZ430-RF TITLE: msp-ez430urf

TI eZ430-RF2500

Schéma FET

● Schémas partie USB (1)

Brochage et E/S

Exemple

Introduction

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

REV: 1.0

Sheet: 1/2

Document Number:

Date: 3/06/2007 03:43:50p



Schémas partie USB (2)

TI eZ430-RF2500

Schéma FET

Schémas partie USB (1)

● Schémas partie USB (2)

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers
Ports série

E/S analogiques

Pilotes de périphériques

GND GND 10 D1 N4148 UCTS UDSR CTS
DSR
DCD
RI/CP
RTS
DTR
TEST0
TEST1
UCC
UCC1
UDD18
GND1
GND2 SIN USB BRXDI BTXDI CLK3410 R21 33k U5 R18 GND4 **GND3** GND2 SN75240PW R23 1k5 SCL E2 MC MSP-eZ430U uss ucc TITLE: msp-ez430u Document Number: REV: 2.0 Date: 5/10/2007 02:01:13p Sheet: 2/2

- p. 7/48



Brochage du MSP430f2274

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Brochage du MSP430f2274Pilotage des Entrées / Sorties

- Mapping mémoire des
- MSP430

 Exemple de port
- (MSP430f149)

 Exemple de port
- (MSP430f149)

Exemple

Introduction

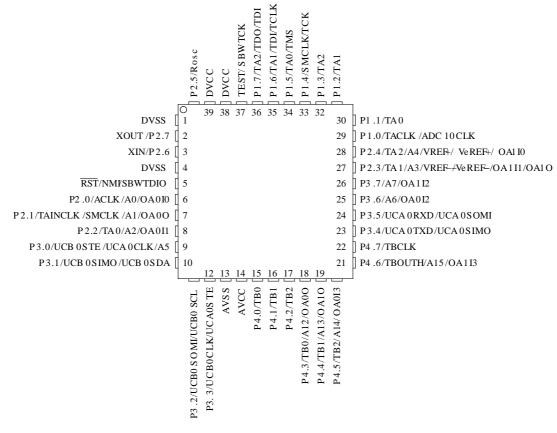
Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques





Pilotage des Entrées / Sorties

TI eZ430-RF2500

Schéma FET

Brochage et E/S

- Brochage du MSP430f2274
- Pilotage des Entrées / Sorties
- Mapping mémoire des
- MSP430
- Exemple de port (MSP430f149)
- Exemple de port (MSP430f149)

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

- Les broches sont multiplexées
 - Un port peut être utilisé comme I/O générique
 - ◆ Il peut correspondre à une fonction (bloc de périphérique)
- Programmation logicielle
 - PxSEL : sélection IO / périphérique
 - ◆ PxIN : registre des données en lecture
 - ◆ PxOUT : registre des données en écriture
 - PxDIR : direction du port
 - Les blocs sont pilotés à des adresses fixes.

- p. 9/48



Mapping mémoire des MSP430

TI eZ430-RF2500

Schéma FET

Brochage et E/S

- Brochage du MSP430f2274
- Pilotage des Entrées / Sorties

Mapping mémoire des MSP430

- Exemple de port (MSP430f149)
- Exemple de port (MSP430f149)

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

- Accès aux registres
- Chaque registre est mappé en mémoire :

```
volatile unsigned char P50UT asm("0x0031");
P5OUT = 0x13;
char *p5 = (char*)0x0031;
*p5 = 0x13;
```



Exemple de port (MSP430f149)

TI eZ430-RF2500

Schéma FET

Brochage et E/S

- Brochage du MSP430f2274
- Pilotage des Entrées / Sorties
- Mapping mémoire des MSP430

Exemple de port (MSP430f149)

Exemple de port

(MSP430f149)

Exemple

Introduction

Gestion du temps

Interruptions

Ports série

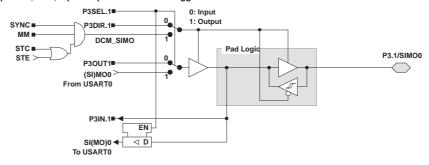
Timers

E/S analogiques

Pilotes de périphériques

input/output schematic (continued)

port P3, P3.1, input/output with Schmitt-trigger



- p. 11/48



Exemple de port (MSP430f149)

TI eZ430-RF2500

Schéma FET

Brochage et E/S

- Brochage du MSP430f2274
- Pilotage des Entrées / Sorties
- Mapping mémoire des MSP430
- Exemple de port
- (MSP430f149)

 Exemple de port

(MSP430f149)

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

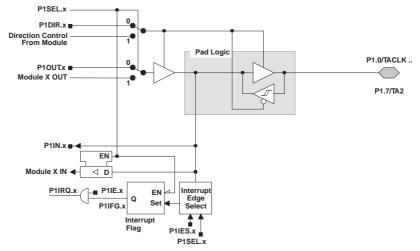
E/S analogiques

Pilotes de périphériques

- Ports 1 et 2 avec interruptions
- Registres PxIFG (flag) et PxIE (interrupt enable)

input/output schematic

port P1, P1.0 to P1.7, input/output with Schmitt-trigger





Exemple simple

TI eZ430-RF2500 Schéma FET

Brochage et E/S

Exemple

Exemple simple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

```
#include <io.h>
#define BIT_GREEN    (1 << 1)
#define BIT_RED     (1 << 0)

void wait(unsigned int n)
{
   int i;
   for(i=0;i<n;i++)
   {
      asm(" nop;\n");
      asm(" nop;\n");
   }
}</pre>
```

```
int main(void)
 unsigned char b;
 // Direction = 0:input 1:output
 // Selector = 0:GPIO 1:peripheral
 P1DIR |= (BIT_GREEN | BIT_RED);
 P1SEL &= ~(BIT_GREEN | BIT_RED);
 P1OUT &= ~(BIT_GREEN | BIT_RED);
 b = 0x01;
 while (0)
   wait(50000); wait(50000);
   wait(50000); wait(50000);
   P1OUT = b;
   b <<= 1;
   if (b == 0x4) b = 0x01;
 return 0;
}
```

- p. 13/48



Présentation des périphériques

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Présentation des périphériques

- Brochage du MSP430f2274
- Schéma bloc du MSP430f2274

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

Le module eZ430-RF2500 est équipé d'un msp430f2274. Ces modèles de msp ont les propriétés suivantes:

- Fréquence interne jusqu'à 16MHz
- TimerA avec 3 registres 16 bits
- TimerB avec 3 registres 16 bits
- USART avec UART + SPI + I2C
- Convertisseurs 10 bits

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

• Présentation des

périphériques

Schéma bloc du

MSP430f2274

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

● Brochage du MSP430f2274

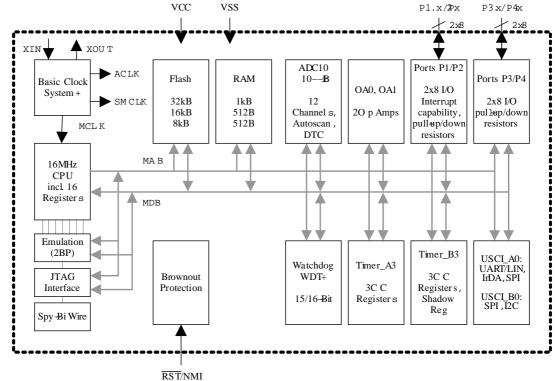
Brochage du MSP430f2274

P1.6/TA1/TDI/TCLK P1.7/TA2/TDO/TDI P1.4/SMCLK/TCK TEST/SBWTCK P1.5/TA0/TMS P1.3/TA2 DVCC 38 37 36 35 34 33 32 DVSS P1 .1/TA0 XOUT/P2.7 2 29 P1.0/TACLK /ADC 10 CLK XIN/P2.6 3 P 2.4/TA 2/A4/VREF+/ Ve REF+/ OAI IO DVSS 27 P2.3/TA1/A3/VREF-/VeREF-/OA1I1/OA1O RST/NMI/SBWTDIO 26 P3 .7/A7/OA1I2 P2 .0/ACLK/A0/OA0I0 25 P3 .6/A6/OA0I2 P2.1/TAINCLK /SMCLK /A1/OA00 24 P3.5/UCA 0RXD/UCA 0SOMI P2.2/TA0/A2/OA0I123 P3.4/UCA 0TXD/UCA 0SIMO $P3.0/UCB\ 0STE\ /UCA\ 0CLK/A5$ P4 .7/TBCLK P3.1/UCB 0SIMO/UCB 0SDA 10 21 P4 .6/TBOUTH/A15/OA1I3 13 14 15 16 17 18 19 P4.4/TB1/A13/OA10 P3. 3/UCB0CLK/UCA0S TE P4.0/TB0 P4.3/TB0/A12/OA00 P 4.5/TB2/A14/OA0I3 P3 .2/UCB0 SOMI/UCB0 SCL P4.1/TB1

- p. 15/48



Schéma bloc du MSP430f2274



TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

- Présentation des périphériques
- Brochage du MSP430f2274

Schéma bloc du MSP430f2274

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Gestion des horloges

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Introduction

Exemple

Gestion du temps

Gestion des horloge

- Gestion des horlog
 Basic Clock +
- Réglage du DCO

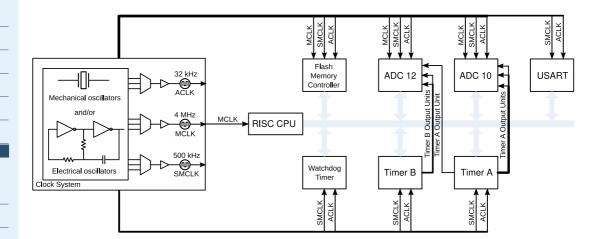
Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

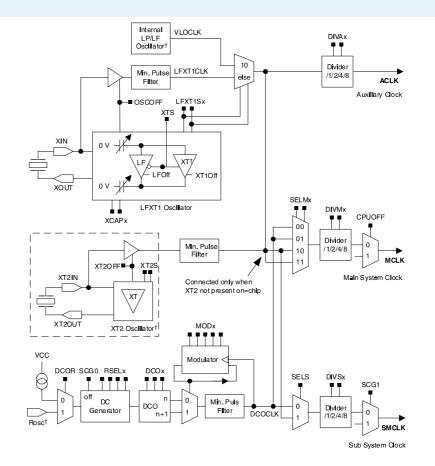


- Un seul bloc d'horloge est utilisé comme référence
- Les blocs doivent être paramétrés en fonction de ces références

- p. 17/48



Basic Clock +



Exemple
Introduction
Gestion du temps
• Gestion des horloges

Racio Clack +

Basic Clock +

Réglage du DCO

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Interruptions

Timers

Ports série

E/S analogiques



TI eZ430-RF2500

Schéma FET

Exemple

Introduction

Interruptions

Timers

Ports série

E/S analogiques

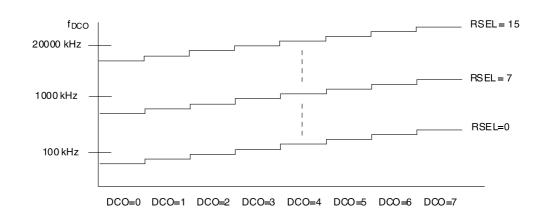
Pilotes de périphériques

Gestion du temps
 Gestion des horloges
 Basic Clock +

Brochage et E/S

Réglage du DCO

Le bloc DCO est programmable à l'aide de son jeu de registre. Les fréquences atteignables couvrent toute la plage de 1.1MHz à 16MHz. La dérive de la fréquence est sensible.



- p. 19/48



Modes de veille

- TI eZ430-RF2500
- Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

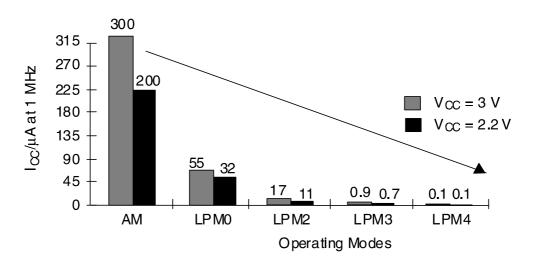
- Modes de veille
- Interruptions
- Ecriture d'un gestionnaire d'interruption

Timers

Ports série

E/S analogiques

- La consommation du MSP430 est gérée grâce à des modes de veille
- Le premier mode permet de couper les instructions
- Les autres modes coupent les autres horloges





Modes de veille

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions Modes de veille

Modes de veille

- Interruptions
- Ecriture d'un gestionnaire d'interruption

Timers

Ports série

E/S analogiques

Pilotes de périphériques

- Macro LPM0,
 - ◆ Modification du bit CPUOFF du registre de contrôle
- La sortie d'un mode de sommeil est faite à la réception d'une interruption (timer)
- LPMO EXIT modifie la valeur du bit CPUOFF dans la sauvegarde de SR sur la pile
- Désactivation partielles des périphériques du MSP
 - ◆ LPM0 : horloge MCLK principale désactivée
 - ◆ LPM1: MCLK et DCO désactivés
 - ◆ LMP2: MCLK, DCO, SMCLK
 -
- Utiliser les interruptions pour contrôler le programme
- Reste la mise en veille des périphériques

- p. 21/48



Interruptions

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

 Modes de veille Modes de veille

Interruptions ■ Ecriture d'un gestionnaire d'interruption

Timers

Ports série

E/S analogiques

- Interruptions à 2 niveaux:
 - GIE dans le registre SR
 - Gestion des sources dans les blocs
- Interruptions à source uniques: l'acquittement est implicite
- Interruptions à sources multiples: acquittement nécessaire
 - ◆ Bits PxIE + PxIES + PxIFG des ports 1 et 2.
 - Comparateurs des registres des timers
 - ...
- Les interruptions sont, par défaut, non ré-entrantes. Le bit GIE est mis à 0 par le matériel à l'entrée d'un gestionnaire d'interruption.



Ecriture d'un gestionnaire d'interruption

```
TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Modes de veille
Interruptions

Ecriture d'un gestionnaire d'interruption

Timers

Ports série
```

E/S analogiques

Pilotes de périphériques

```
//interrupt (PORT1_VECTOR) port1_irq_handler(void)
#pragma vector=PORT1_VECTOR
__interrupt void
port1_irq_handler(void)
{
    if (P1IFG & (P1IE & (1 << 2)))
        {
        SWITCH_RED_LED();
    }
    P1IFG = 0;
}</pre>
```

- p. 23/48



Timers

П	eZ4	30-	RF	250	00

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

TimersTimer A

- Timer B
- Timers et modes
- Timers et modes (2)
- Timers et modes (3)

Ports série

E/S analogiques

- Utilisation d'une horloge de référence par bloc
- Comparateurs multiples et gestion des interruptions
- Interruptions
 - acquittement automatique (TxCCR0)
 - acquittement logiciel (TxIV)
 - ◆ Les interruptions sont priorisées
- Les timers contiennent des registres sur 16 bits
- Les différents modes permettent d'avoir une gestion efficace du temps

Timer A

TI eZ430-RF2500 Schéma FET

Brochage et E/S

Exemple Introduction

Gestion du temps

Interruptions

Timers

Timers

• Timer A

Timer B

● Timers et modes

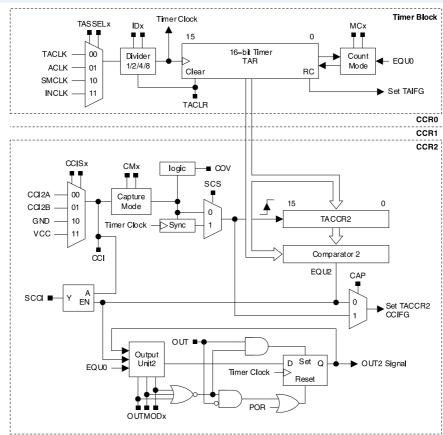
Timers et modes (2)

Timers et modes (3)

Ports série

E/S analogiques

Pilotes de périphériques



- p. 25/48



Timer B

TI eZ430-RF2500 Schéma FET Brochage et E/S Exemple Introduction Gestion du temps Interruptions Timers Timers Timer A

● Timer B

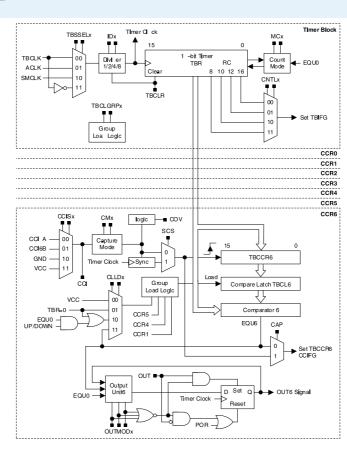
● Timers et modes

Timers et modes (2)

● Timers et modes (3)

Ports série

E/S analogiques



Timers et modes

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Gestion du temps

Interruptions

Timers

Timers

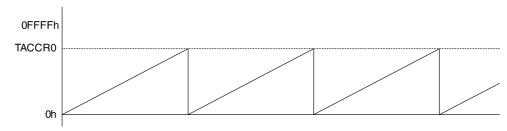
- Timer ATimer B
- Timers et modes
- Timers et modes (2)Timers et modes (3)

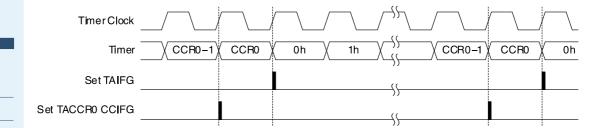
Ports série

E/S analogiques

Pilotes de périphériques

■ Le mode UP est sans doute le plus utilisé.





- p. 27/48



Timers et modes (2)

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps
Interruptions

Timers

- Timers
- Timer ATimer B
- Timers et modes

• Timers et modes (2)

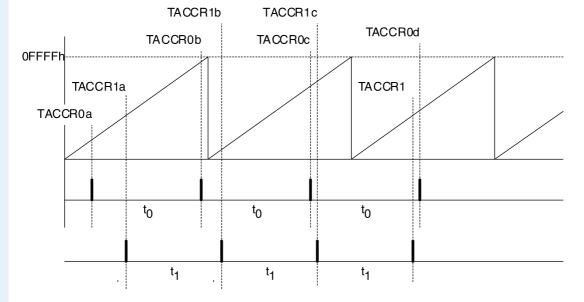
Timers et modes (3)

E/S analogiques

Ports série

Pilotes de périphériques

■ "Continuous Mode"



Timers et modes (3)

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Fuerrale

Exemple

Gestion du temps

Introduction

Interruptions

Timers

Timers

- Timer ATimer B
- Timers et modes
- Timers et modes (2)

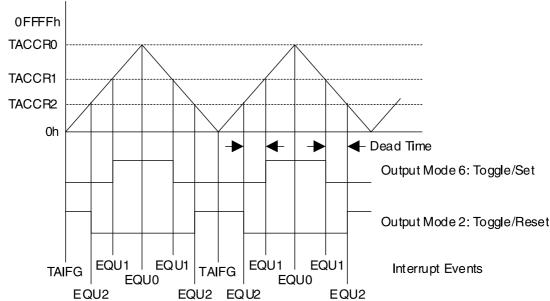
Timers et modes (3)

Ports série

E/S analogiques

Pilotes de périphériques





- p. 29/48



Ports série

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

- Ports série
- Ports série

E/S analogiques

- Plusieurs modèles disponibles
 - Certains sont en mode exclusif
 - Modules USI (SPI, I2C)
 - Modules USART (UART, SPI, I2C)
 - ◆ Ceux du 2274 sont doubles (Modules USCI)
 - USCI_Ax (UART + IrDA + SPI)
 - USCI_Bx (SPI + I2C)
- Les ports séries peuvent se piloter par interruption
- Le débit est calculé à partir d'une horloge de référence
- Le débit est dépendant des horloges externes

USCI A

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

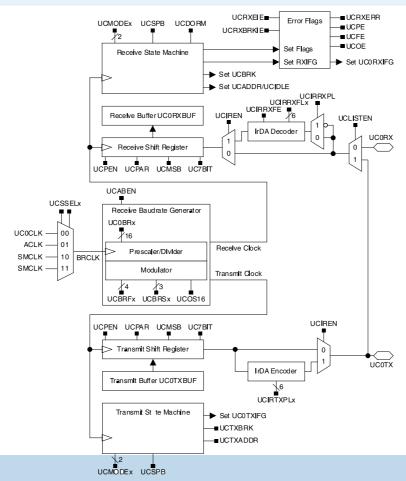
Ports série

USCI A

Ports série

E/S analogiques

Pilotes de périphériques



- p. 31/48

Ports série

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Ports série

Ports série

• USCI A

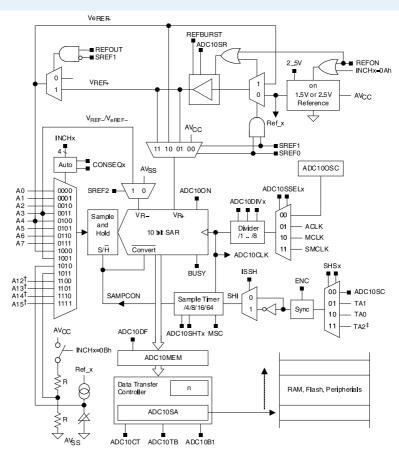
● Ports série

E/S analogiques

- Les paramètres à utiliser pour la modulation sont à choisir dans les tables fournies par le manuel.
- La fréquence de référence ne doit pas changer.
- Le choix de la fréquence de référence est donc important.
- Si beaucoup d'erreurs apparaissent il est conseillé de passer à un code correcteur d'erreur logiciel.

XX ADC 10





- p. 33/48



ADC 10

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

TI eZ430-RF2500

Ports série

■ ADC 10

- La programmation de l'ADC reste dans le modèle que pour les autres composants:
 - Choisir une fréquence d'échantillonage par rapport à une fréquence de référence
 - Choisir un canal à numériser
 - Attendre la stabilisation
 - Demander la lecture
 - Attendre la validation
 - ◆ Possibilité de recevoir une interruption au besoin



Programmation de pilotes

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

Programmation de pilotes

- Périphériques et schématique
- schéma bloc
- smartRF
- États du CC2500
- state machine
- paquet
- SPI
- command Strobe
- Écriture de pilotes
- Pilote d'émission en attente

- Un pilote de périphérique doit utiliser les ressources de la plateforme.
- Une attention spéciale doit être prise pour ne pas trop simplifier le matériel.
- Une configuration *globale* de la plateforme est obligatoire pour tirer le meilleur de l'application.
- Les données des applications peuvent être utilisés pour simplifier les sources et réduire la complexité du code.

- p. 35/48



Périphériques et schématique

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers Ports série

E/S analogiques

Pilotes de périphériques

Programmation de pilotes Périphériques et schématique

schéma bloc

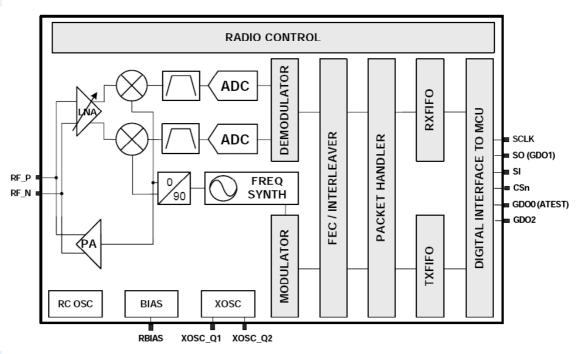
- smartRF
- États du CC2500
- state machine
- paquet • SPI
- command Strobe
- Écriture de pilotes
- Pilote d'émission en attente

- 1. La schématique n'est que le point de départ.
 - connexions entre les composants
 - low level (bit) communications
- 2. Il faut manipuler également le comportement des composants.



Schéma du CC2500

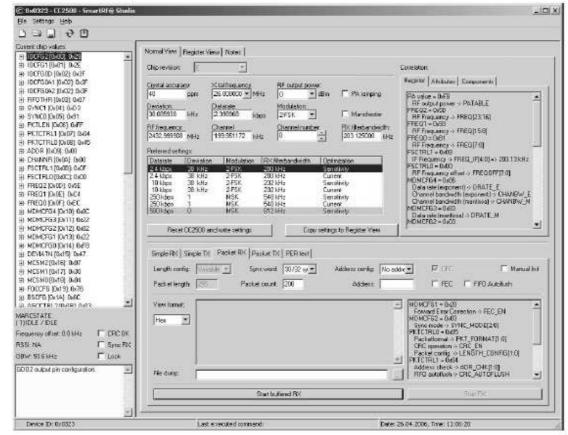
TI eZ430-RF2500 Schéma FET Brochage et E/S Exemple Introduction Gestion du temps Interruptions Timers Ports série E/S analogiques Pilotes de périphériques Programmation de pilotes • Périphériques et schématique ● schéma bloc smartRF ● États du CC2500 state machine paguet SPI command Strobe Écriture de pilotes • Pilote d'émission en attente



- p. 37/48



SmartRF



TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple Introduction

Gestion du temps

Interruptions Timers

Ports série

E/S analogiques

- Pilotes de périphériques
- Programmation de pilotes
- Périphériques et schématique
- schéma bloc

● smartRF

- États du CC2500
- state machine paquet
- SPI
- command Strobe
- Écriture de pilotes
- Pilote d'émission en attente



États du CC2500

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

E/S analogiques

Ports série

Pilotes de périphériques

- Programmation de pilotes
- Périphériques et schématique
- schéma bloc
- smartRF

● États du CC2500

- state machine
- paquetSPI
- command Strobe
- Écriture de pilotesPilote d'émission en attente
- Dilata diferiacian avec inte (

- États
- **■** Transitions
- Temps / latences
- Consommation

- p. 39/48

Lowest power mode. Most register values are retained



Machine d'état du CC2500

Figure E. Cimplified state died

Typ. current consumption 400nA, or 900nA when wake-on-radio (WOR) is enabled. Sleep Default state when the radio is no receiving or transmitting. Typ current consumption: 1.5mA. Idle Used for calibrating frequency synthesizer upfront (entering receive or transmit mode can All register values are Manual freq. Crystal retained. Typ. current consumption; 0.16mA. oscillator off then be done aulcker) Transitional state. Typ. current consumption: 7.4mA. Frequency synthesizer startup, Frequency synthesizer is turned on, can optionally be calibrated, and then settles to the correct frequency. Transitional state. Typ. current consumption: 7.4mA. Frequency synthesizer is on, ready to start transmitting. Transmission starts very quickly after receiving the STX command strobe. Typ. current consumption: 7.4mA. optional calibration settling Frequency synthesizer o radio (WOR) Typ. current Typ. current consumption: 11.1mA at -12dBm output, 15.1mA at -6dBm output, consumption: from 13.3mA (strong input signal) to 16.6mA Transmit mode Receive mode 21.2mA at 0dBm output. (weak input signal) Optional transitional state. Typ In FIFO-based modes, transmission is turned off and this state entered if the TX FIFO becomes empty in the middle of a packet. Typ. current consumption: 1.5mA In FIFO-based modes, reception is turned off and this state entered if the RX FIFO overflows. Typ. current consumption: 1.5mA. current consumption: 7.4mA TX FIFO underflow Optional freq. synth. calibration Idle

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

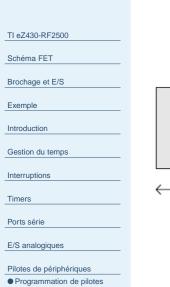
- Programmation de pilotes
- Périphériques et schématique
- schéma bloc
- smartRF
- États du CC2500

● state machine

- paquet
- SPI
- command Strobe
- Écriture de pilotes
- Pilote d'émission en attente
- Dilata d'ámission avac intr. /1



format du paquet du CC2500



		\leftarrow	0	——Optional data whitening———Optionally FEC encoded/decoded— ptional CRC-16 calculation———	→ •
Preamble bits (10101010)	Syncword	Length field	Address field	Data field	CRC-16
← 8 x <i>n</i> bits →	≺ 16/32 bits	× _{bits} >	< ⁸ bits≯	<8 x n bits>	←16 bits→

- p. 41/48

Legend:

Inserted automa

processed and r

Optional user-pr

processed but no

Unprocessed us and/or whitening

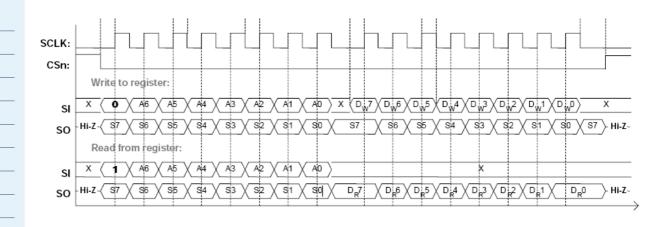


• Périphériques et schématique

schéma bloc smartRF ● États du CC2500 state machine paquet

command Strobe Écriture de pilotes Pilote d'émission en attente

Programmation du CC2500 par port SPI



TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

Pilotes de périphériques

- Programmation de pilotes
- Périphériques et schématique
- schéma bloc
- smartRF
- États du CC2500
- state machine
- paquet

● SPI

- command Strobe
- Écriture de pilotes
- Pilote d'émission en attente



Command Strobe

TI eZ430-RF2500 Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers Ports série

E/S analogiques

Pilotes de périphériques

- Programmation de pilotes
- Périphériques et schématique
- schéma bloc
- smartRF
- États du CC2500
- state machine
- paguet SPI

command Strobe

- Écriture de pilotes
- Pilote d'émission en attente

Address	Strobe Name	Description
0x30	SRES	Reset chip.
0x31	SFSTXON	Enable and calibrate frequency synthesizer (if MCSM0 , FS_AUTOCAL=1). If in RX (with CCA): Go to a wait state where only the synthesizer is running (for quick RX / TX turnaround).
0x32	SXOFF	Turn off crystal oscillator.
0x33	SCAL	Calibrate frequency synthesizer and turn it off (enables quick start). SCAL can be strobed in IDLE state without setting manual calibration mode (MCSM0 , FS_AUTOCAL=0)
0x34	SRX	Enable RX. Perform calibration first if coming from IDLE and MCSM0 . FS_AUTOCAL=1.
0x35	STX	In IDLE state: Enable TX. Perform calibration first if MCSM0 • FS_AUTOCAL=1. If in RX state and CCA is enabled: Only go to TX if channel is clear.
0x36	SIDLE	Exit RX / TX, turn off frequency synthesizer and exit Wake-On-Radio mode if applicable.
0x38	SWOR	Start automatic RX polling sequence (Wake-on-Radio) as described in Section 19.5.
0x39	SPWD	Enter power down mode when CSn goes high.
0x3A	SFRX	Flush the RX FIFO buffer. Only issue in IDLE, TXFIFO_UNDERFLOW or RXFIFO_OVERFLOW states.
0x3B	SFTX	Flush the TX FIFO buffer. Only issue in IDLE, TXFIFO_UNDERFLOW or RXFIFO_OVERFLOW states.
0x3C	SWORRST	Reset real time clock.
0x3D	SNOP	No operation. May be used to pad strobe commands to two bytes for simpler software.

- p. 43/48



Écriture de pilotes

TI eZ430-RF2500

Schéma FET

Brochage et E/S

Exemple Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

- Pilotes de périphériques Programmation de pilotes
- Périphériques et schématique
- schéma bloc
- smartRF
- États du CC2500
- state machine
- paquet
- SPI

command Strobe

Écriture de pilotes

- Pilote d'émission en attente

- L'écriture d'un pilote doit suivre les principes suivants:
- Contrôle de la machine à état depuis le logiciel.
- Utilisation de toutes les fonctionnalités matérielles.
- Fournir une abstraction aux autres parties de l'application. Envoyer un paquet sur la radio ne doit pas être plus compliqué qu'un appel de fonction.



Pilote d'émission en attente

```
TI eZ430-RF2500
Schéma FET
Brochage et E/S
Exemple
Introduction
Gestion du temps
Interruptions
Timers
Ports série
E/S analogiques
Pilotes de périphériques

    Programmation de pilotes

• Périphériques et schématique
• schéma bloc
smartRF
● États du CC2500
state machine
paquet
SPI
```

```
/* pkt < 64 bytes, wait EOP */
void cc1100_utx(char *buffer, uint8_t length)
  cc1100_tx_error = 0;
  cc1100_check_fifo_xflow_flush();
  /* Fill tx fifo */
  CC1100_SPI_TX_FIFO_BYTE (length);
  CC1100_SPI_TX_FIFO_BURST (buffer, length);
  /* Send packet and wait for complete */
  cc1100_gdo2_set_signal(CC1100_GDOx_SYNC_WORD);
  CC1100_HW_GDO2_DINT();
  CC1100_SPI_STROBE(CC1100_STROBE_STX);
  while (! ( CC1100_HW_GD02_READ() )); /* GD02 goes high = SYNC TX */
  while ( CC1100_{HW}_{GDO2}_{READ}() )); /* GDO2 goes low = EOP
  CC1100_HW_GDO2_EINT();
```

- p. 45/48



command Strobe Écriture de pilotes

Pilote d'émission avec intr. (1)

TI eZ430-RF2500 Schéma FET

Brochage et E/S

Exemple

Introduction

Gestion du temps

Interruptions

Timers

Ports série

E/S analogiques

- Programmation de pilotes
- Périphériques et schématique schéma bloc
- smartRF
- États du CC2500
- state machine
- paquet SPI
- command Strobe
- Écriture de pilotes
- Pilote d'émission en attente

```
void cc1100_tx(char *buffer, uint8_t length)
 uint8_t txbytes; /* bytes free in the fifo */
 uint8_t tosend;
 cc1100_check_fifo_xflow_flush();
  cc1100_tx_packet = buffer;
  cc1100_tx_length = length;
  cc1100_tx_offset = 0;
  cc1100 tx ongoing = 1;
  cc1100_tx_error = 0;
  cc1100\_tx\_sent = 0;
  /* Fill tx fifo */
  CC1100_SPI_TX_FIFO_BYTE (length);
  TX_WRITE_BLOCK();
  /* Send packet but don't wait complete */
  CC1100_HW_GDO2_DINT();
  CC1100_HW_GDO0_DINT();
  cc1100_gdo0_set_signal(CC1100_GDOx_TX_FIFO); /* gdo0 tx fifo
  cc1100_gdo2_set_signal(CC1100_GDOx_SYNC_WORD); /* gdo2 sync & eop
                                                      /* want an irq for EOP
  CC1100_HW_GDO2_IRQ_ON_DEASSERT();
  CC1100 HW GDOO IRO ON DEASSERT();
                                                      /* want an irg on Tx < thr */
  CC1100_SPI_STROBE(CC1100_STROBE_STX);
                                                      /* start
  while (! ( CC1100_HW_GDO2_READ() ));
                                                     /* GDO2 high = sync TX
  CC1100_HW_GDO0_EINT();
                                                                            */
                                                      /* allow irg on gdo0
  CC1100_HW_GDO2_EINT();
                                                       /* allow irq on gdo2
```



Pilote d'émission avec intr. (2)

```
TI eZ430-RF2500
Schéma FET
Brochage et E/S
Exemple
Introduction
Gestion du temps
Interruptions
Timers
Ports série
E/S analogiques
Pilotes de périphériques

    Programmation de pilotes

• Périphériques et schématique
• schéma bloc
smartRF
● États du CC2500
state machine
```

```
void cc1100_tx_pkt_data()
 uint8_t txbytes; /* bytes free in the fifo */
 uint8_t tosend;
  CC1100 SPI ROREG(CC1100 REG TXBYTES, txbytes);
  txbytes = 63 - txbytes; /*room free in Tx FIFO */
  tosend = MIN(txbytes, cc1100_tx_length);
  CC1100_SPI_TX_FIFO_BURST(cc1100_tx_packet, tosend);
  cc1100_tx_packet += tosend;
  cc1100_tx_length -= tosend;
/* called when done on asynchroneous Tx */
void cc1100_tx_done_intr(void)
  if (cc1100_check_fifo_xflow_flush())
      cc1100_tx_error = 1;
  cc1100_tx_ongoing = 0;
  cc1100_tx_sent
  CC1100 HW IRO PACKET ASSERT();
```

- p. 47/48



paquetSPI

command StrobeÉcriture de pilotesPilote d'émission en attente

Pilote d'émission avec intr. (3)

```
TI eZ430-RF2500
Schéma FET
Brochage et E/S
Exemple
Introduction
Gestion du temps
Interruptions
Timers
Ports série
E/S analogiques
Pilotes de périphériques
Programmation de pilotes
• Périphériques et schématique
schéma bloc
● smartRF
● États du CC2500
state machine
```

paquetSPI

command Strobe

Pilote d'émission en attente

```
void cc1100_interrupt_handler(uint8_t pin)
  int interrupt_policy;
  if (pin == CC1100 GDO0)
    interrupt_policy = cc1100_gdo0_cfg;
  else if (pin == CC1100_GDO2)
    interrupt_policy = cc1100_gdo2_cfg;
    return;
  switch (interrupt_policy)
    case CC1100 GDOx TX FIFO:
                                    /* Tx FIFO threshold */
      cc1100_tx_pkt_data();
    case CC1100_GDOx_RX_FIFO:
                                    /* Rx FIFO threshold */
      cc1100_rx_pkt_data();
    case CC1100_GDOx_SYNC_WORD:
      if (cc1100_tx_ongoing == 1) /* Tx EOP */
        cc1100_tx_done_intr();
      else if (cc1100_rx_ongoing) /*Rx\ EOP\ */
        cc1100_rx_pkt_data();
                                    /* Rx SYNC */
      else
        cc1100_rx_pkt_start();
    case CC1100 GDOx CHIP RDY:
    default:
      break ?
```