

Analog Interface Circuit (AIC) TLC32044

Processamento Digital de Sinal I
Paulo Marques e Artur Ferreira
ISEL, 2002

Características 1/3

- Tecnologia CMOS
- ADC e DAC no mesmo *chip*
- As conversões AD e DA podem ter diferentes ritmos: síncronas ou assíncronas
- Frequência de amostragem programável, com o valor máximo de 19.2 KHz
- Gama dinâmica de ± 1.5 ou $\pm 3V$

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

2

Características 2/3

- Filtros *anti-aliasing* e de reconstrução programáveis (condensadores comutados) filtros passa-baixo de 8ª ordem;
- O filtro *anti-aliasing* é precedido por um filtro contínuo;
- O filtro de reconstrução é seguido de um filtro contínuo;
- Efectua correcção $\frac{\text{sen}(x)}{x}$ de 2ª ordem;

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

3

Características 3/3

- Quantização com resolução de 14 bit
- Codificação em código dos complementos:

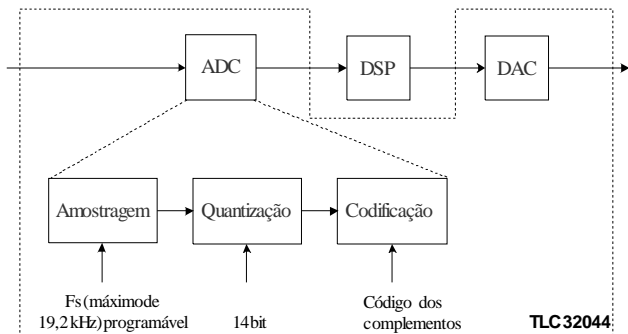
$$\{-2^{13}, \dots, 0, \dots, 2^{13} - 1\}$$

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

4

Integração com o DSP

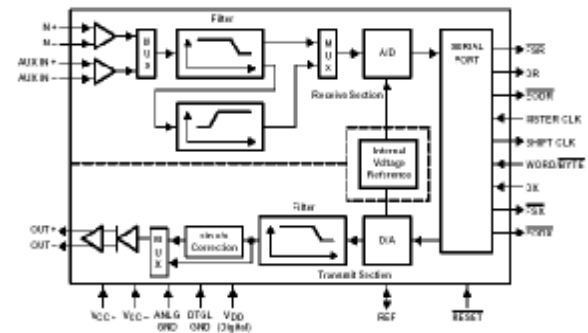


20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

5

Diagrama de Blocos

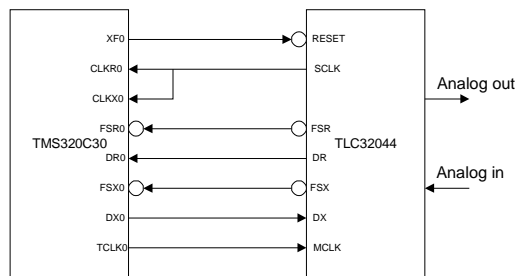


20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

6

Interface entre TMS320C30 e TLC32044



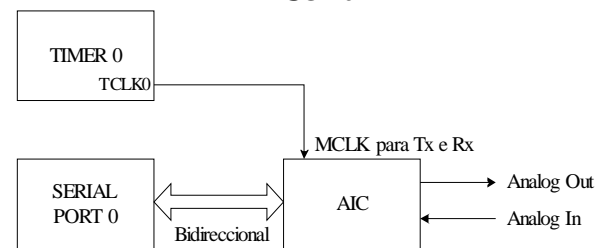
- Reset: EVM faz reset usando XF0 flag
- AIC faz output de sincronismo por FSX e FSR e sinais de clock-shift SCLK para o EVM
- EVM dá clock ao AIC através do TCLK0, pino do TIMER0

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

7

Interface entre TMS320C30 e TLC32044



$$MCLK = \frac{XTAL}{4} = \frac{30}{4} = 7.5MHz$$

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

8

Programação AIC

Registo	Endereço	Designação
TR	0x808048h	Transmit register
RR	0x80804ch	Receive register

Secondary transmission – O DSP pode configurar o AIC via TR
A/D data é a 14 bits; os 2 lsb's dizem o que segue:

- 00 – A próxima palavra é input data
- 11 – A próxima palavra é *secondary transmission*

Secondary transmission:

- 00 – A palavra descreve registo A
- 10 – A palavra descreve registo B

TA e RA são as palavras transmit receive register para A
TB e RB são as palavras transmit receive register para B

Podem existir frequências diferentes de transmissão e recepção

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

9

Programação AIC

Transmissão normal
14 bits em complemento para 2 0 0

Segue transmissão secundária
14 bits em complemento para 2 1 1

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

X X TA X X RA 0 0

X TB X RB 1 0

X X X X X X X X d9 d8 d7 d6 d5 d4 d3 d2 1 1

Os registos A e B definem a frequência de amostragem f_s e a frequência dos condensadores comutados f_{SCF} .

	Nome	Estado	Acção
TA e TB são recarregados após cada conversão D/A	d2	1	A/D Passa-alto activo
	d3	0	Loopback inactivo
	d4	0	Usar entrada analógica primária
RA e RB são recarregados após cada conversão A/D	d5	1	Transmissão/recepção síncronos
	d6,d7	0,1	1V, 1.5V
	d9	1	Correcção Sin x/x activa

20/11/2002

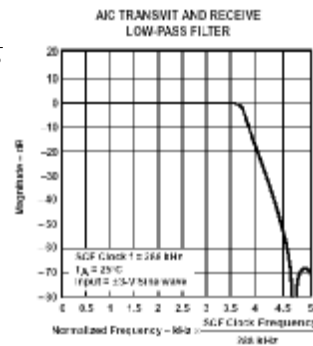
Paulo Marques e Artur Ferreira Nov. 2002

10

Programação AIC

$$f_{SCF} = \frac{f_{Mclk}}{2A} \quad f_s = \frac{f_{Mclk}}{2AB}$$

- A frequência de relógio é $f_{Mclk}=7.5MHz$
- f_{SCF} é a frequência dos condensadores comutados; tipicamente é 288 kHz
- f_s é a frequência de amostragem sendo submúltipla de f_{SCF}



20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

11

Programação AIC

- Quando F_{SCF} não é 288kHz, a resposta em frequência, vem escalada, na frequência pelo quociente desse valor por 288kHz;
- Programar o AIC consiste em:
 1. Iniciar TIMER 0
 2. Fazer RESET ao AIC
 3. Iniciar SERIAL PORT 0
 4. Retirar o AIC de RESET
 5. Iniciar o AIC

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

12

Sysinit.asm (rotina aicinit)

```

aicinit: call wait_transmit_0      | call wait_transmit_0
        ldi 3,r1                  | ldi 3,r1
        sti r1,*+ar0(8)           | sti r1,*+ar0(8)
        call wait_transmit_0      | call wait_transmit_0
        ldi 1a34h,r1 ;counter A   | ldi 2a7h,r1 ;control register
        sti r1,*+ar0(8)           | sti r1,*+ar0(8)
        ldi *+ar0(12),r1          | ldi *+ar0(12),r1
                                        xor if,if ;clear out all interrupt flags
                                        or @enbl_sp0_r,ie ;enable serial port 0
                                        rets
        call wait_transmit_0      |
        ldi 3,r1                  |
        sti r1,*+ar0(8)           |
        call wait_transmit_0      |
        ldi 4892h,r1 ;counter B   |
        sti r1,*+ar0(8)           |
        ldi *+ar0(12),r1          |

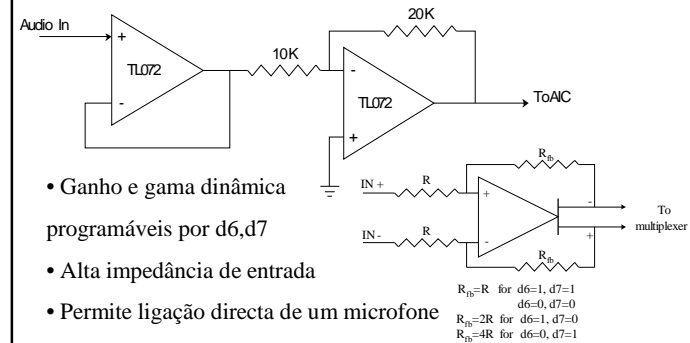
wait_transmit_0:
        xor if,if
        ;wait for the transmit interrupt
wloop:  tstb 10h,if ;flag to be set.
        bz wloop
        rets
    
```

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

13

Interface Analógica Entrada (Rx)



- Ganho e gama dinâmica programáveis por d6,d7
- Alta impedância de entrada
- Permite ligação directa de um microfone

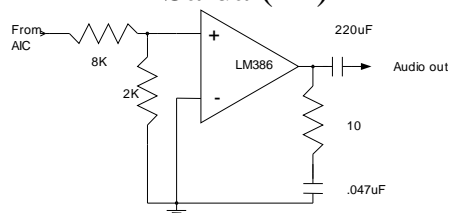
$R_{fb}=R$ for d6=1, d7=1
 $R_{fb}=0$ for d6=0, d7=0
 $R_{fb}=2R$ for d6=1, d7=0
 $R_{fb}=4R$ for d6=0, d7=1

20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

14

Interface Analógica Saída (Tx)



- Ganho global de 4 (o LM386 tem ganho 20)
- Saída analógica com gama dinâmica [-5,+5] Volts
- Permite ligação directa a um *speaker* (cargas com baixa impedância)

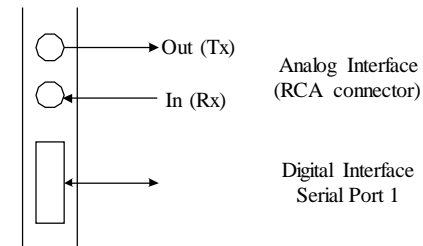
20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

15

Interface exterior na placa

Um canal analógico e um canal digital



20/11/2002

Paulo Marques e Artur Ferreira Nov. 2002

16