

Architecture des ordinateurs - IE.1102 - A1 - 2021-2022 - Semestre 2

Sans documents

Sans calculatrice

Pas de points négatifs. Durée 1h30

Documentation nécessaire en fin de sujet.

1° en complément 补码: 范围为 [-128, 127]
 正数补码为其二进制本身
 负数补码为其反码 +1 (注意位数)
 反码: 反转所有位

A] Nombre et arithmétique :

Q1) Quel est le codage du chiffre 42 en complément à 2 sur 8 bits représenté en hexa?

- a) 0x2A b) 0x42 c) 0xD6 d) 0x42

十六进制: A~F = 10~15 反码: 最高位为符号位(0正1负) 负数除符号位取反

Q2) Quel est le codage du chiffre -21 en complément à 2 sur 8 bits représenté en hexa?

- a) 0xEB b) 0x21 c) 0x15 d) 0xF1

Q3) Quel est l'état des Flags NZVC après le calcul 145 + 11 sur 8 bits ?

- a) 1001 b) 1000 c) 0001 d) 0010

N: 当数 [0, 127], N=0, 否则 N=1
 Z: 当数为 0, Z=1, 否则 Z=0
 V: 当溢出 V=1, 否则 V=0
 C: 当进位 or 借位, C=1, 否则 C=0

Q4) Quelle est la représentation décimale du chiffre %11000011 codé en complément à 2 sur 8 bits ?

- a) 33 b) 195 c) -60 d) -61 保留(-1)

负数补码 = 反转二进制所有位 +1

Q5) Quel chiffre faut-il ajouter à 100 pour faire -100 (codage sur 8 bits complément à 2).

- a) 200 b) 56 c) 128 d) 9

Q6) On veut coder des chiffres entiers entre -10.0 et +9,9 (précision au dixième combien faut-il de bits au minimum pour leur représentation ?

2^8

- a) 8 b) 9 c) 10 d) 11

Q7) On lit un convertisseur A/D 12 bits récupérant une température entre -200° et +200° à pleine échelle, quelle est environ la précision de la mesure ?

1 英寸位 + 2" 数据

- a) 1° b) 0,5° c) 0,1° d) 0,05°

Q8) Quel est le codage du chiffre -22 sur 16 bits en complément à 2 ?

- a) 0x0016 b) 0xFF16 c) 0xFFEA d) 0xFFDE

B] Architecture / composants

Q9) Qu'est ce qui peut être aléatoire dans une RAM ?

- a) Le contenu
- b) L'adresse**
- c) Le fonctionnement
- d) La donnée

RAM - L'adresse

Q10) Dans la technologie FLASH

- a) Il est impossible d'écrire une donnée
- b) Il est impossible de lire une donnée
- c) Le contenu est perdu lors des coupures d'alim
- d) L'écriture se fait par blocs**

分区写入

Q11) Un processeur dispose d'un bus d'adresse de 24 bits et d'un bus de donnée de 32 bits (4 octets). Quelle est la taille de l'espace adressable ?

- a) 16 Mo
- b) 32 Mo
- c) 64Mo**
- d) 1282 Mo

pour le bus d'adresse de 24 bits 地址数: 2^{24}

寻址空间大小: 可寻址地址数量 × 每地址存储数据量

Q12) Quel composant consomme le plus d'énergie par bit stocké ?

- a) Disque dur
- b) registre**
- c) Mémoire FLASH
- d) Mémoire SDRAM

Q13) Quelle proposition est vraie ?

- a) Un registre est plus rapide d'une SRAM**
- b) Une DRAM est plus rapide qu'une SRAM
- a) Une FLASH est plus rapide qu'une DRAM
- b) Une DRAM utilise 4 transistors par bit stocké

Q14) Dans l'architecture de Von Neuman...

小结

- a) Les périphériques lisent la mémoire
- b) Les périphériques sont lus directement par le processeur**
- c) La mémoire gère les écritures
- d) Le bus gère les écritures

Q15) A quoi sert le registre R14 du processeur ARM ?

- a) Il contient toujours le résultat des opérations arithmétiques et logiques
- b) A retenir l'adresse du code en cours d'exécution en mémoire
- c) C'est un registre d'usage général
- d) A retenir l'adresse de retour d'un sous-programme appelé avec l'instruction BL**

Q16) Lors de l'instruction : LDR

R0,[R1]
↑
加锁位
值存放在寄存器中

L'opérande source est :

- a) Dans le processeur
- c) Dans un périphérique
- b) Dans la mémoire**
- d) Dans la pile

MOV ASR 右移

右移最高位填低位

C] Assembleur ARM

LSL 左移 0 填高位

左移并
相等于×2

ldr 存入 orr 或 and 与 ero ⊕ 同或
sub 减 adds 不带进位加 adcs 带进位加

str 存入 地址

strh 半存入 (低位)
并清零高位

Pour les questions suivantes **les registres sont initialisés à 0**. Le contenu des mémoires est aléatoire avant qu'on écrive dedans, et la mémoire autours de l'adresse 0 est de type RAM (lecture / écriture). **ARM架构中：寄存器32位，以补码形式存储** 取值范围 $[0, 2^{32}-1]$

Q17) Quel est le contenu du registre R0 après : **mov**
 R₁=0
 a) 0x5 b) 0xF5 c) 0xfffffffffb d) 0xFFFFFFFF

Q18) Quel est le contenu du registre R0 après : **orr**
 R₁=0
 a) -3 b) 0 c) 3 d) 0xFD

Q19) Quel est le contenu de R3 après

ldr
sub
减法
r4,=8
r3,r4,#3
R₁=3
a) 5 b) -5 c) -3 d) 3

Q20) Quel est le contenu du registre R0 après :

and
R₁=4
R₀=4 与运算
a) 0x14 b) 0x0E c) 0x00000002 d) 0

Q21) Quel est le contenu de R0 après :

ldr
sub
减法
r0,=0x19
r0,r0,r0
R₁=0
a) 0xFFFFFE7 b) 0xFFFFFFFED c) 0 d) 0x19

Q22) Quel est le contenu de r0 après :

ldr r0,=0x40000000
adds r0,r0,r0
adds r0,r0,r0
R₁=0
溢出
a) 0 b) 0x80000000 c) 0xc0000000 d) E0000000

Q23) Quel est le contenu de r0 après :

ldr r7,=0x80
adds r7,r7,r7
adcs r0,r7,r7
 转为2进制计算
 $r7' = 2r7$
 $r0 = 2 \cdot r7' = 4r7$
 然后换为16进
 a) 0x201 b) 0x200 c) 0x100 d) 0x80

Q24) Quelle est la valeur du registre R0 après

```
ldr r0,=0x30
ldr r1,=0x20
sub r0,r0,r1
```

- a) 0xFFFFFFFF0 b) 0 c) 0x10 d) 0x20

Q25) Quel est le contenu de r0 après : (EOR est l'opération de ou exclusif logique)

```
ldr r0,=0x12345678
ldr r1,=0x87654321
eor r0,r1,r0
```

\oplus 同0异1

- a) 0x022444220 b) 0x95511559 c) 0x9775579 d) 0x77777777

Q26) Quelle est la valeur du registre R0 après

```
ldr r0,=0x87654321
mov r0,r0,asr #1
```

右移一位
算术右移一位
将位最高位值复制到新的最高位
符号位填充左空位

- a) 0x08765432 b) 0x76543210 c) 0x0ECA8642 d) 0xC3B2A190

Q27) Quel est le contenu de R0 après : (GT : Greater or Than)

```
mov r0,#10
cmp r0,#8 R0与#8比较,由于大于,标志位GT(greater than会被设置)
sublt r0,r0,#2
```

且只有LT(less than)被设置时执行减法(sub)操作

- a) 10 b) 0x10 c) 8 d) 6

Q28) Quel est le contenu de r0 après :

```
mov r0,#0x41
add r0,r0,r0,lsl #2
```

加法
左移一位 \Leftrightarrow 将其*2
左移2位(右侧0填充)

- a) 0x20 b) 0x82 c) 0x145 d) 0x249

Q29) On souhaite faire $r0 * 3$ quel est la ligne qui convient ?

- a) add r0,r0,r0 lsl #1 左移一位 \Leftrightarrow 将其*2
 $n_0 + 2n_1 = n_0^*3$ b) add r0,r0,r0 lsl #5
 $= n_0^*3$ c) add r0,r0,r0 lsr #1
 $= n_0^*3$

Q30) Quelle est la valeur du registre R0 après :

```
ldr r1,=0x12345678
str r1,[r0] 将R1存入R0内存地址
ldr r0,[R0,3]
```

R0中存储顺序为:0X78, 0X56, 0X34, 0X12
 地址: 0X00, 0X01, 0X02, 0X03
 ;ldrb : Load byte

R0中读取偏移3的字节
 存入

- a) 0x12 b) 0x34 c) 0x56 d) 0x78

Q31) Quelle est la valeur du registre R0 après :

ldr r1,=0x12345678
strh r1,[r8] 将 r1 的半字值(低16位)存入 r8 的低16位位置 ; strh ; store half word
ldr r0,[r8] ; (ldr : load word)
 ↪ 32位 (高字节已清零)

- a) 0x1234 b) 0x5678 c) 0x12345678 d) 0x56781234

Q32) Quelle est la durée d'exécution du programme suivant en cycle horloge ?

ldr r0,=3 ②
mov r0,r1 ① r1 复制至 r0 中
sub r0,r1,r2 ① r0 = r1 - r2
bal ici ③ 跳转到 ici
ici add r0,r0,#3 ① r0 = r0 + 3
 ; bal : branch always

ldr/bal ③
bne

- a) 5 b) 7 c) 8 d) 9

Q33) Quelle est la durée d'exécution du programme suivant :

Ici mov r0,#5 ①
subs r0,r0,#1 ①
bne ici ③
Fin ; (Branch if not equal to 0)

- a) 5 b) 15 c) 21 d) 34

Q34) Quel est le contenu de R0 après :

Ldr r0,=0x12345678
Mov r0,r0,asr #3

- a) 0x2468ACF b) 0x00012345 c) 0x45678000 d) 0x91A2B3C0

Q35) Coder l'instruction **RSBLE R4,R5,#3** 0011

- a) 0xD25450003 b) 0xE25450003 c) 0xD26450003 d) 0xD26540003

Q36) Coder l'instruction **ORRS R7,R2,R3,LSL R5**

- a) 0xE3927513 b) 0xE1927513 c) 0xE1927515 d) 0xE1727513

Q37) Coder l'instruction **ANDLTS R10,R10,R10,ROR#10**

- a) 0xB01AA566 b) 0xB0199569 c) 0xB02BB56B d) 0xB01AA56A

D) Péphériques

Q38) On programme le port parallèle de la façon suivante : bits 0 en entrée, bit 1-15 en sortie
Quelle valeur place-t-on dans PC2 ?

- a) 0x0001 b) 0xFFFF c) 0xFF00 d) 0x000F

Q39) On transmet 1Koctets de données sur 8 bits sans parité à 9600 bauds. Quel est l'ordre de grandeur du temps de transmission ?

$$\begin{aligned} \text{1 octet} &= 10 \text{ bits} \\ \text{1 baud} &= 1 \text{ bit} \end{aligned}$$

$= 9600 \text{ bits/s} \approx 1000 \text{ bits/s}$
 $\approx 1000 \text{ octets/s} = 1000 \text{ octets/s}$

- a) 1 seconde b) 10 secondes c) 20 minutes d) 2 heures

Q40) Quelle est la valeur du bit de parité paire de la donnée 0x52?

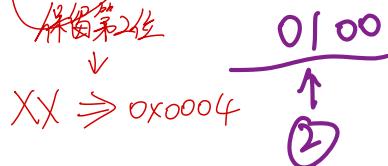
- a) 0 b) 1 c) 00 d) 11

包括数据
位数有偶数个

奇数个 ⇒ 偶数等于1

Q41) On souhaite tester un interrupteur placé sur le bit 2 du port parallèle. Que vaut XX ?

Ldrh r0,[PD]
Ands r0,#XX
Bne INTER_ON



- a) 0 b) 1 c) 2 d) 4

0000 0100

Q42) On souhaite communiquer avec un module Bluetooth, quel interface faut-il utiliser ?

- a) Série b) Timer c) convertisseur A/D d) Parallèle

Codage d'une instruction arithmétique

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cond	0	0	#	op	S	Rn		Rd		operand2																					

Les champs variables ont la signification suivante :

Cond : condition d'exécution de l'instruction. Les différentes valeurs possibles sont les suivantes :



Condition (cond)		Code Opération (op)					
Valeur	condition	Valeur	condition	Valeur	condition	Valeur	condition
0000	EQ	1000	HI	0000	AND	1000	TST
0001	NE	1001	LS	0001	EOR	1001	TEQ
0010	CS	1010	GE	0010	SUB	1010	CMP
0011	CC	1011	LT	0011	RSB	1011	CMN
0100	MI	1100	GT	0100	ADD	1100	ORR
0101	PL	1101	LE	0101	ADC	1101	MOV
0110	VS	1110	{AL}	0110	SBC	1110	BIC
0111	VC	1111	NV	0111	RSC	1111	MVN

: Format du 2nd opérande. S'il s'agit d'une valeur immédiate, ce champ est à 1. Sinon, il est à 0.

S : est à 1 si les codes conditions doivent être mis à jour.

Rn : numéro du registre opérande 1

Rd : numéro du registre destination

operand2 : second opérande. Trois formats sont possibles :

Opérande **registre, décalage éventuel spécifié par une constante**:

11	10	9	8	7	6	5	4	3	2	1	0
#shift	Sh	0	Rm								

#shift : 3 Nombre de positions de décalage

Sh : ASR Type de décalage (00 = LSL, 01 = LSR, 10 = ASR, 11 = ROR)

Rm : RS Numéro du registre

Opérande **registre, décalage éventuel spécifié par un registre**:

11	10	9	8	7	6	5	4	3	2	1	0
Rs	0	Sh	1	Rm							

Rs : Numéro du registre contenant le nombre de positions de décalage

Sh : Type de décalage (00 = LSL, 01 = LSR, 10 = ASR, 11 = ROR)

Rm : Numéro du registre

Opérande immédiat (de la forme valeur_8_bits * 2^2n) : (Bit 25 à 1)

11	10	9	8	7	6	5	4	3	2	1	0
#rot	value										

#rot : n

value : Valeur sur 8 bits

Codage d'une instruction de branchement

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cond		1	0	1	L	offset																									

Les champs variables ont la signification suivante :

cond : condition du branchement. Les différentes valeurs possibles sont les mêmes que pour les instructions arithmétiques (voir ci-dessus).

L : Si l'adresse suivante doit être mémorisée dans r14 (adresse de retour), ce champ est à 1. Sinon, il est à 0.

offset : déplacement (d) divisé par 4. L'adresse cible du branchement sera calculée comme :
adresse cible = adresse du branchement (PC) + 8 + d

Périphérique rappel

Le sens des échanges se programme par les registres PC0 PC1 et PC2 du périphérique parallèle.

La lecture ou l'écriture du port parallèle s'effectue par le registre PD.

Pour chaque bit, le sens des échanges est programmé comme suit :

PC0	PC1	PC2	Etat
1	0	1	Sortie
1	0	0	Entrée

Temps d'exécution des instructions :

ALU	1	+1 si shift +2 si Rd est pc
B, BL, BX	3	
LDR	3	+2 si rd est PC

$$Q1) 42 \Rightarrow \begin{array}{c} 001010 \\ \hline 1 \quad 2 \quad 4 \quad 8 \quad 16 \quad 32 \\ 64 \quad 128 \end{array}$$

$$Q2) -21 \Rightarrow \begin{array}{c} 10010101 \\ \downarrow \\ 11010101 + 1 \\ \hline 11101011 \end{array}$$

$$\begin{array}{r} 14 \\ 11 \\ \downarrow \\ E \quad B \end{array}$$

$$Q3) 145 + 11 = 156 = 10011110$$

$$\begin{array}{l} \downarrow \\ \text{超}(0, 127) \quad \text{未溢出, 未借位} \\ N=1 \quad N=0 \quad C=0 \\ \text{不为0: } Z=0 \end{array} \quad \left. \begin{array}{l} \\ \\ \end{array} \right\} NZVC = 1000$$

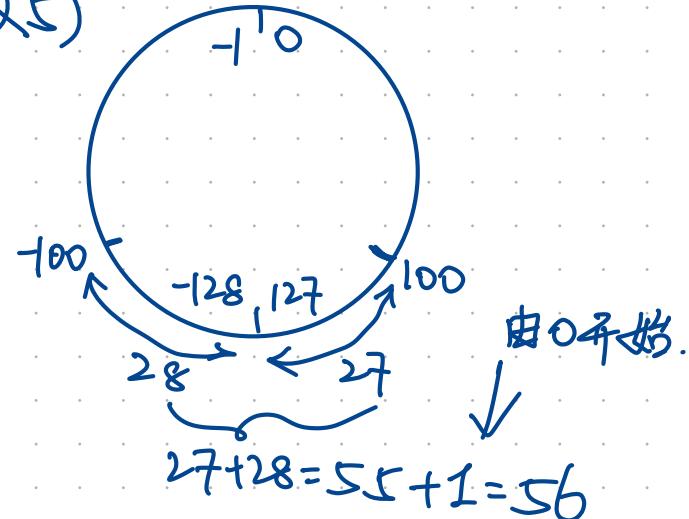
$$Q4) 11000011$$

↓ en complément: 反转所有位

$$(-1) 00111100 + 1$$

$$(-1) \begin{array}{c} 00111101 \\ \hline 32 \quad 16 \quad 8 \quad 4 \\ + + + + \end{array} = -61$$

Q5)



十六进制

	A	B	C	D	E	F
10	10	11	12	13	14	15

Q6) pour faire en entiers: $\begin{array}{c} -100 \sim +100 \\ (-10.0) \quad (+9.9) \end{array}$

8位: $[-128, 127] \supseteq [-100, +100]$

7位: $[-64, +63] \not\supseteq [-100, +100]$

Q7) 12 bit $\Rightarrow (-2'', 2'') = (-2048, 2047)$

$\begin{array}{c} \downarrow \\ (-200^\circ, +200^\circ) \end{array} \quad \left. \begin{array}{l} \text{末一位} \\ \rightarrow \text{小数点后一位} \end{array} \right\}$

$$Q8) -22 = \underbrace{000000000}_{\downarrow \text{complément}} \underbrace{0010110}_{\begin{array}{l} 111111111101001 + 1 \\ \hline 111111111101010 \\ \downarrow 16 \quad 10 \\ F \quad F \quad E \quad A \end{array}}$$

Q11) pour le bus d'adresse de 24 bits \Rightarrow 地址数: 2^{24}

可寻址空间大小: 可寻址地址数量 \times 每地址存储数据量

$$= 2^{24} \times 4$$

$$= 2^{26}$$

$$1MB = 1024kB = 2^{10}$$

$$2^{24} \rightarrow 16MB$$

Q17) $-5 \Rightarrow \dots \dots \dots 0000\ 0101$
 R: $1010 + 1 \rightarrow 1011 = 11$
 $\begin{array}{cccccc} F & F & F & F & F & F \\ \underline{4} & \underline{4} & \underline{4} & \underline{4} & \underline{4} & \underline{4} \end{array}$

Q18) $R1 \text{ } 0x0000\ 0000$
 $\#3 \text{ } 0x0000\ 0003$ } or $R_0 : 00000003$

Q19) $R3 = R4 - \#3$
 $= 8 - 3 = 5$

Q20) $R3 : 0x0000\ 0000$
 $\#14: 0x00001110$ } $\Rightarrow R_0 = 0$

Q22) $\Rightarrow R_0 = 0x40000000 + 0x40000000 = 0x80000000$
 $\Rightarrow R_0 = 0x80000000 + 0x80000000 = 0x(\cancel{16})\underline{00000000} = 0$
 16进制中进0

Q23) $r7 = 0x80$
 $r7 = 0x80 + 0x80 = 1000\ 0000 + 1000\ 0000 = \cancel{10000\ 0000} = 0x100$
 $\begin{array}{r} 10000\ 0000 \\ + 10000\ 0000 \\ \hline 10\ 0000\ 0000 \end{array} = 0x200$

$$Q24) \begin{array}{r} r_0 = 0 \times 30 = 110000 \\ r_1 = 0 \times 20 = 100000 \\ \hline - \\ \hline 10000 \end{array}$$

$$r_0 = 0 \times 10$$

$$Q25) r_0 = 0 \times 12345678 = 0001\ 0010\ 0011\ 0100\ 0101\ 0110\ 0111\ 1000$$

$$\begin{array}{r} r_1 = 0 \times 87654321 = 1000\ 0111\ 0110\ 0101\ 0100\ 0011\ 0010\ 0001 \\ \hline 1001\ 0101\ 0101\ 0001\ 0001\ 0101\ 0101\ 1001 \\ = 0 \times 9\ 5\ 5\ 1\ 1\ 5\ 5\ 9 \end{array}$$

$$Q26) r_0 = 0 \times 87654321$$

$$\begin{aligned} &= 1000\ 0111\ 0110\ 0101\ 0100\ 0011\ 0010\ 0001 \\ &= 1100\ 0011\ 1011\ 0010\ 1010\ 0001\ 1001\ 0000 \\ &= 12\ 3\ 11\ 2\ 10\ 1\ 9\ 0 \\ &= C3B2A190 \end{aligned}$$

$$Q28) r_0 = 0 \times 41$$

$$= 0100\ 0001$$

$$r'_0 = 1st \#2 \Rightarrow 0000\ 0100$$

$$\begin{array}{r} r_0 + r'_0 = \\ 0001\ 0100\ 0001 \\ + \\ 0000\ 0100 \\ \hline 0001\ 0100\ 0101 \end{array}$$

$$= 0 \times 145$$

Q31) $r_1 = 0x12345678$
 $\Rightarrow 0x5678$

[r8] $0x1000$

\downarrow

[r8] new $0x5678$

Q32) $r_0 = 3$ (3)
 $r_0 \leftarrow r_1$ (1)
 $r_0 = r_1 - r_2$ (1)
bal ici (3)
 $r_0 = r_0 + 3$ (1)

Q33) ~~$r_0 = 5$ (1) ici (3)~~
 ~~$r_0 = r_0 - 1 = 4$ (1) ici (3)~~
 ~~$r_0 = 3$ (1) ici (3)~~
 ~~$r_0 = 2$ (1) ici (3)~~
 ~~$r_0 = 1$ (1) ici (3)~~
 $r_0 = 0$ (1)

$(1+3) \times 5 + 1 = 21$

Q34) $r_0 = 0x12345678$
 $= 0001\ 0010\ 0011\ 0100\ 0101\ 0110\ 0111\ 1000$
 $0000\ 0010\ 0100\ 0110\ 1000\ 1010\ 1100\ 1111$
0 2 4 5 8 A C F

Q1) $42 \Rightarrow 101010$ $\begin{array}{r} 32 \\ 16 \\ 8 \\ 4 \\ 2 \\ 1 \end{array}$

$$\Rightarrow \frac{0010}{2} \quad \frac{1010}{10}$$

$$\Rightarrow 0x2A$$

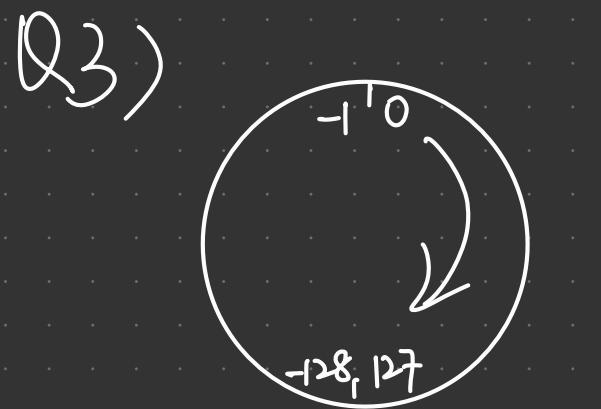
A	B	C	D
10	11	12	13

Q2) $-21 \Rightarrow 00010101$

$$\Rightarrow 11101010 + 1$$

$$= \frac{11101011}{15} \quad \frac{11}{11}$$

$$\Rightarrow 0xE8$$



$$145 + 11$$

$$= 156 > 127 \quad N=1 \quad Z=0$$

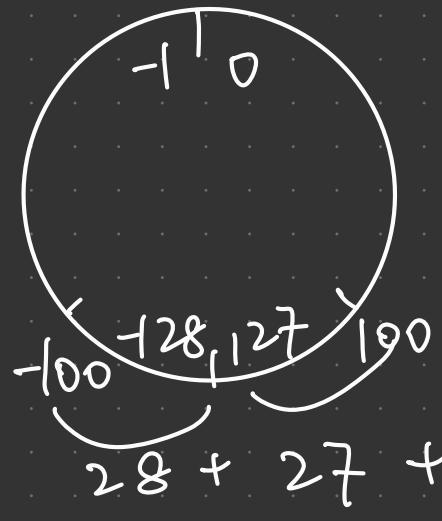
$$V=0 \quad C=0$$

Q) 11000011
负数

(-1) 00111100 + 1

$$\begin{array}{r} 00111101 \\ \underline{-32} \quad \underline{16} \quad \underline{8} \quad \underline{4} \quad 1 \\ = -61 \end{array}$$

Q5)



$$\begin{array}{r} 128 \quad 64 \quad 32 \quad 16 \quad 8 \quad 4 \quad 2 \quad 1 \\ \sum^8 \quad - \quad - \quad - \quad - \quad - \quad - \end{array}$$

$$28 + 27 + 1 = 56$$

Q6) -100 ~ 99

$$\left(-2^8, 2^8 \right)$$

Q7) 12 bits \Rightarrow 1 符号位 + 2"

$$\pm 2'' = \underline{\left(-2048, 2048 \right)}$$

200,0

Q8) -22 = 0000...0001 0110

$$\overline{\rightarrow 111\ldots 1110 \ 1001 + 1}$$

$$\begin{array}{ccccccc} & \overline{111\ldots 1110} & \overline{1001} & & & & \\ = & 0 & F & E & \overset{10}{A} & & \end{array}$$

$$Q11) 2^{24} \times 2^2 = 16 \times 4 = 64 M_{\odot}$$

$$Q17) -5 \Rightarrow 0101$$

$$\begin{array}{r} 1010+1 \\ \hline 1011 \end{array}$$
$$\Rightarrow 0x F \dots F B$$

$$Q18) R1=0+3=3$$

$$Q19) r_4=8$$

$$r_3=r_4-3=5$$

$$Q20) R0=0$$

$$Q21) r_0=0x19$$

$$r_0=0$$

$$Q22) r_0=80000000$$

$$r'_0 = \cancel{16}0000000 = 0$$

进位0.

$$Q23) r_7=10000000$$

$$r'_7=10000000$$

$$r_0 = \frac{1}{2} \frac{0000}{0} \frac{0000}{0}$$

$$Q24) R_0=00110000$$

$$\begin{array}{r} - R_1=00100000 \\ \hline 00010000 \end{array}$$

$$n=0x10$$

$$Q25) r0 = \underline{0001} \underline{0010} \underline{0011} \underline{0100} \underline{0101} \underline{0110} \underline{0111} \underline{1000}$$

$$\oplus \quad \underline{1000} \quad \underline{0111} \quad \underline{0110} \quad \underline{0101} \quad \underline{0100} \quad \underline{0011} \quad \underline{0010} \quad \underline{0001}$$

$$\begin{array}{cccccccc} & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ \hline & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{array}$$

$$= 0x \underset{9}{\cancel{1}} \underset{5}{\cancel{1}} \underset{5}{\cancel{1}} \underset{5}{\cancel{1}} \underset{9}{\cancel{1}}$$

$$Q26) r0 = \underline{1000} \quad \underline{0111} \quad \underline{0110} \quad \underline{0101} \quad \underline{0100} \quad \underline{0011} \quad \underline{0010} \quad \underline{0001}$$

$$\begin{array}{cccccccc} & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ & 1 & 2 & 3 & 1 & 1 & 2 & 0 \\ \hline & 1 & 2 & 3 & 1 & 1 & 2 & 0 \end{array}$$

$$= 0xc3b2a190$$

$$Q27) r0 = 10$$

$$cmp : GT = 1 \quad LT = 0$$

$$r0 = 10 = 10$$

$$Q28) r0 = 0x4f$$

$$= 0100 \quad 0001$$

$$\Rightarrow \underline{0000} \quad \underline{0100}$$

$$\begin{array}{cccc} & 1 & 0 & 1 & 0 \\ & 0 & 0 & 1 & 0 \\ & 0 & 0 & 0 & 1 \\ \hline & 0 & 0 & 1 & 0 \end{array}$$

$$= 0x45$$

$$Q30) r1 = 0x12345678$$

$$0x78 \quad 0x56 \quad 0x34 \quad 0x12$$

$$\begin{array}{cccc} 00 & 01 & 02 & 03 \end{array}$$

$$Q31) r8 = 5678$$

$$r0 = 0x00005678$$

$$Q34) r0 = \underline{0001} \underline{0010} \underline{0011} \underline{0100} \underline{0101} \underline{0110} \underline{0111} \underline{1000}$$

$$= 02468ACF$$

$$Q35) \frac{1101}{13} \quad \frac{0010011}{2} \quad \frac{1}{7} \quad \frac{0101}{5} \quad \frac{0100}{4} \quad \underline{\underline{00011}} \quad 10 \quad 0101$$

$$Q36) \frac{1110}{E} \frac{000}{1} \frac{1100}{9} \frac{1}{2} \frac{0010}{7} \frac{0111}{3} \frac{0011}{0} \frac{000}{0} \frac{0101}{}$$

$$Q37) \frac{1011}{B} \frac{001}{2} \frac{0000}{1} \frac{1}{A} \frac{1010}{A} \frac{1010}{A} \frac{1010}{A} \frac{11}{1010}$$

$$Q38) 0101 \quad 0010$$

(+1)