

ALGO
QCM

1. Un graphe peut être ?
 - (a) Orienté
 - (b) Non orienté
 - (c) A moitié orienté
 - (d) Désorienté
2. Un graphe partiel G' de $G = \langle S, A \rangle$ est défini par ?
 - (a) $\langle S, A' \rangle$ avec $A' \subseteq A$
 - (b) $\langle S', A \rangle$ avec $S' \subseteq S$
 - (c) $\langle A', S' \rangle$ avec $A' \subseteq S$ et $S' \subseteq A$
3. Dans un graphe non orienté, s'il existe une chaîne reliant x et y pour tout couple de sommet $\{x, y\}$ le graphe est ?
 - (a) complet
 - (b) partiel
 - (c) parfait
 - (d) connexe
4. Deux arêtes d'un graphe non orienté sont dites adjacentes si ?
 - (a) il existe deux arêtes les joignant
 - (b) le graphe est incomplet
 - (c) le graphe est valorisé
 - (d) elles ont au moins une extrémité commune
5. Dans un graphe orienté, toute chemin d'un sommet vers lui-même est ?
 - (a) non élémentaire
 - (b) élémentaire
 - (c) Un circuit
 - (d) Un cycle
 - (e) Une chaîne
6. Dans un graphe orienté, le sommet x est adjacent au sommet y si ?
 - (a) Il existe un arc (x, y)
 - (b) Il existe un arc (y, x)
 - (c) Il existe un chemin (x, \dots, y)
 - (d) Il existe un chemin (y, \dots, x)

7. Dans un graphe non orienté G , un graphe partiel G' de G est une composante connexe du graphe G ?
- (a) Vrai
 - (b) Faux
8. Un graphe G défini par le triplet $G = \langle S, A, C \rangle$ est ?
- (a) étiqueté
 - (b) valué
 - (c) valorisé
 - (d) numéroté
9. Un sous-graphe G' de $G = \langle S, A \rangle$ est défini par ?
- (a) $\langle S, A' \rangle$ avec $A' \subseteq A$
 - (b) $\langle S', A \rangle$ avec $S' \subseteq S$
 - (c) $\langle A', S' \rangle$ avec $A' \subseteq S$ et $S' \subseteq A$
10. Un graphe G non orienté connexe est un graphe complet ?
- (a) oui
 - (b) non



QCM N°4

lundi 25 octobre 2021

Question 11

Soient X et Y deux variables aléatoires finies entières indépendantes, de fonctions génératrices

$$G_X(t) = \frac{2t+1}{3} \quad \text{et} \quad G_Y(t) = \frac{t+2}{3}$$

- a. $P(X+Y=2) = \frac{2}{9}$
- b. $P(X+Y=2) = \frac{1}{9}$
- c. $P(X+Y=1) = \frac{4}{9}$
- d. $P(X+Y=1) = \frac{6}{9}$
- e. Aucun des autres choix

Question 12

Soit la série entière $\sum (-1)^n x^n$ et notons R son rayon de convergence.

Considérons sa fonction somme, définie pour tout $x \in]-R, R[$ par : $f(x) = \sum_{n=0}^{+\infty} (-1)^n x^n$

- a. $R = -1$
- b. Pour tout $x \in]-R, R[$, $f(x) = \frac{1}{1-x}$
- c. Pour tout $x \in]-R, R[$, $f(x) = \frac{1}{1+x}$
- d. Pour tout $x \in]-R, R[$, $f(x) = e^{-x}$
- e. Aucun des autres choix

Question 13

Soit une série entière $\sum a_n x^n$ et notons R son rayon de convergence.

Considérons sa fonction somme, définie pour tout $x \in]-R, R[$ par : $f(x) = \sum_{n=0}^{+\infty} a_n x^n$

- a. Pour tout $x \in]-R, R[$, $\int_0^x f(t) dt = \sum_{n=1}^{+\infty} n a_n x^{n-1}$
- b. Pour tout $x \in]-R, R[$, $\int_0^x f(t) dt = \sum_{n=0}^{+\infty} a_n \frac{x^{n+1}}{n+1}$
- c. Aucun des autres choix

Question 14

Soit une série entière $\sum a_n x^n$ telle que $\frac{a_{n+1}}{a_n} \xrightarrow{n \rightarrow +\infty} -\frac{1}{3}$. Le rayon de convergence de la série vaut :

- a. $R = 3$
- b. $R = -3$
- c. $R = \frac{1}{3}$
- d. $R = -\frac{1}{3}$
- e. Aucun des autres choix

Question 15

Soit X une variable aléatoire prenant ses valeurs dans \mathbb{N} , c'est-à-dire que $X(\Omega) = \mathbb{N}$. On suppose connues les probabilités $P(X=n)$ pour tout $n \in \mathbb{N}$.

- a. La série $\sum P(X=n)$ converge et $\sum_{n=0}^{+\infty} P(X=n) = 1$
- b. $P(X \geq 5) = \sum_{n=5}^{+\infty} P(X=n)$
- c. $P(X \geq 5) = 1 - \left(\sum_{n=0}^4 P(X=n) \right)$
- d. $P(X \geq 5) = 1 - \left(\sum_{n=0}^5 P(X=n) \right)$
- e. Aucun des autres choix.

Question 16

Soient $q \in]0, 1[$ et X une variable aléatoire entière telle que $G_X(t) = \frac{(1-q)t}{1-qt}$

- a. $P(X=2) = (1-q)$
- b. $P(X=2) = (1-q) \times q$
- c. $P(X=2) = (1-q) \times q^2$
- d. Aucun des autres choix

Question 17

Soit X une variable aléatoire entière de fonction génératrice $G_X(t) = \frac{te^t}{e}$.

Son espérance vaut :

- a. $E(X) = 0$
- b. $E(X) = 1$
- c. $E(X) = \frac{1}{e}$
- d. Aucun des autres choix

Question 18

Soient E un espace vectoriel sur \mathbb{R} et $\mathcal{F} = \{e_1, \dots, e_n\}$ une famille de E .

Cette famille est libre si et seulement si :

- a. $\forall x \in E, \exists (\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$ tel que $x = \lambda_1 e_1 + \dots + \lambda_n e_n$
- b. $\exists (\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n, \lambda_1 e_1 + \dots + \lambda_n e_n = 0_E$
- c. $\forall (\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n, \lambda_1 e_1 + \dots + \lambda_n e_n = 0_E \implies \lambda_1 = \dots = \lambda_n = 0$
- d. Aucun des autres choix

Question 19

Dans l'espace vectoriel $E = \mathbb{R}^3$, considérons la famille $\mathcal{F} = \{(1, 1, 0), (0, 1, 1)\}$.

- a. Cette famille est libre
- b. Cette famille est génératrice de \mathbb{R}^3
- c. Aucun des autres choix.

Question 20

Dans l'espace vectoriel $E = \mathbb{R}^2$, considérons la base $\mathcal{B} = \{(1, 2), (3, 4)\}$.

Soit le vecteur $u = (3, 5)$. Pour trouver ses coordonnées dans la base \mathcal{B} , on cherche $(\lambda_1, \lambda_2) \in \mathbb{R}^2$ tel que :

- a.
$$\begin{cases} \lambda_1 + 2\lambda_2 = 3 \\ 3\lambda_1 + 4\lambda_2 = 5 \end{cases}$$
- b.
$$\begin{cases} \lambda_1 + 3\lambda_2 = 3 \\ 2\lambda_1 + 4\lambda_2 = 5 \end{cases}$$
- c. Aucun des autres choix.

QCM 4 Azar Chap20 D (conds1-3pp427-430 ex 3,5,10)

Choose the one best answer for the situations.

21. You do not think you can afford to go to the NBA game in Paris tonight. Which do you say?
- a. If I had a lot of money, I will go to the game tonight.
 - b. If I had a lot of money, I would go to the game tonight.
 - c. If I have a lot of money, I would have gone to the game tonight.
 - d. If I have a lot of money, I would go to the game tonight.
22. What happens if it rains?
- a. If it rains, the streets get wet.
 - b. If it rains, the streets are getting wet.
 - c. If it is raining, the streets getting wet.
 - d. If it rains, the streets would get wet.
23. Somebody stops you on the way to the NBA game at the Accord Arena. You say:
- a. If you went right at the end of this street, you see the Arena on your left.
 - b. If you go right at the end of this street, you will see the Arena on your left.
 - c. If you went right at the end of this street, you will see the Arena on your left.
 - d. If you go right at the end of this street, you seen the Arena on your left.
24. You are not going abroad this year. You say: "If I ____ abroad, I'd have to speak English."
- a. go
 - b. went
 - c. would go
 - d. had gone
25. Jean Pierre has forgotten his book every day this week. If he ____ it again today, I will not allow him into class.
- a. have
 - b. won't have
 - c. didn't have
 - d. doesn't have
26. I wanted to send my parents an email last night but I didn't have enough time. In other words:
- a. If I had enough time, I would have sent them an email.
 - b. If I hadn't enough time, I would have sent them an email.
 - c. If I had had enough time, I would have sent them an email.
 - d. If I had had enough time, I would send them an email.

27. Choose the sentence with no mistakes.

- a. If I were rich, I would buy a new car.
- b. If I were rich, I would have buy a new car.
- c. If I was rich, I would bought a new car.
- d. If I were rich, I will buy a new car.

28. The seminar would only have been held if...

- a. less than half the participants cancelled
- b. the participants had cancelled
- c. more than half the participants hadn't cancelled
- d. none of the participants hadn't cancelled

29. Bill wants to change jobs because he does not make enough money. Which sentence matches?

- a. If Bill had made more money, he would have stayed in this job.
- b. If Bill made more money, he stays in his current job.
- c. If Bill got more money, he would not think about finding another job.
- d. All of the above.

30. Which of the following is the **only** sentence with no mistakes?

- a. I will buy the stock only if interesting rates go down.
- b. I will have bought the stock only if interest rates go down.
- c. I will buy the stock only if interest rates goes down.
- d. I will buy the stock only if interest rates go down.

Q.C.M n°4 de Physique

41- Une distribution de charges sphérique crée au point M un potentiel électrique $V(\theta, \phi)$, on peut donc affirmer que le vecteur champ électrique s'écrira :

- a) $\vec{E} \begin{pmatrix} 0 \\ E_\theta \\ E_\phi \end{pmatrix}$ b) $\vec{E} \begin{pmatrix} E_r \\ 0 \\ E_\phi \end{pmatrix}$ c) $\vec{E} \begin{pmatrix} E_r \\ E_\theta \\ 0 \end{pmatrix}$ d) $\vec{E} \begin{pmatrix} 0 \\ 0 \\ E_\phi \end{pmatrix}$

42- Soit la fonction potentiel électrique $V(r) = r \cdot e^{-\frac{a}{r}}$,
(a est une constante).

Le champ électrique qui dérive de ce potentiel sera d'expression :

- a) $\vec{E} = e^{-\frac{1}{r}} \left(1 - \frac{a}{r}\right) \cdot \vec{u}_r$ b) $\vec{E} = -e^{-\frac{1}{r}} \left(1 + \frac{a}{r}\right) \cdot \vec{u}_r$ c) $\vec{E} = e^{-\frac{a}{r}} \cdot \vec{u}_r$

43- Un champ électrostatique \vec{E} est dit entrant lorsqu'il est créé par :

- a) Un proton b) Un neutron c) Un électron

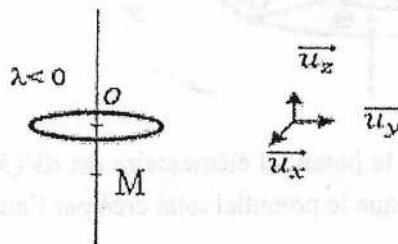
44- Soit un anneau de rayon R et d'axe (Oz), chargé avec une densité linéique λ supposée constante et positive. La charge élémentaire dQ d'un élément de longueur dl de l'anneau s'exprime par :

- a) $dQ = \lambda d\theta$ b) $dQ = \lambda dR$ c) $dQ = \lambda R d\theta$

45- On considère le système chargé de la question (44). La charge totale de l'anneau est d'expression :

- a) $Q = 2\pi R \lambda$ b) $Q = 2\pi \lambda$ c) $Q = \pi R \lambda$

46- On considère un anneau dans le plan (x,y) centré en O et chargé uniformément avec une densité linéique $\lambda < 0$ (voir la figure ci-dessous).



Par symétrie, le vecteur champ électrique créé en M ($z < 0$) est porté par :

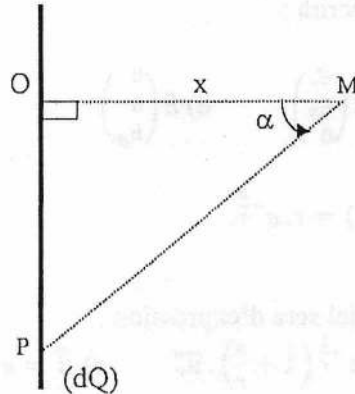
- a) $-\vec{u}_z$
b) $+\vec{u}_z$
c) Aucune des deux réponses précédentes n'est correcte

47- Un électron envoyé entre deux armatures d'un condensateur plan est soumis à une force électrique \vec{F}_e qui vérifie :

- a) Orthogonale aux armatures et orientée de la plaque (+) vers la plaque (-)
b) Parallèle aux armatures
c) Orthogonale aux armatures et orientée de la plaque (-) vers la plaque (+)

48- On montre qu'un élément de longueur situé en P d'un fil de charge linéique constante λ crée un champ électrique \vec{dE} en un point M extérieur au fil, de composante : $dE_x(x) = \frac{k\lambda}{x} \cos(\alpha) d\alpha$.

L'angle α est tel qu'indiqué ci-dessous, on pose : (OM = x).



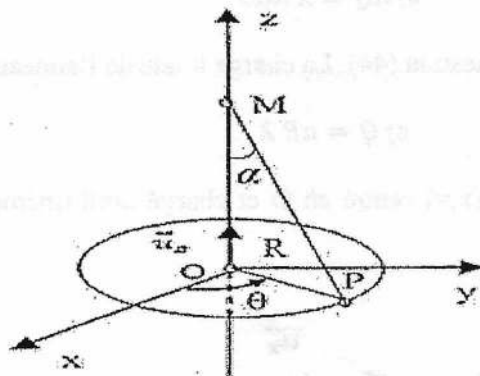
Le champ électrique total créé par le fil infini s'écrit :

a) $E(x) = \frac{k\lambda}{x}$

b) $E(x) = \frac{2k\lambda}{x}$

c) $E(x) = 2\sin(\alpha) \frac{k\lambda}{x}$

49- Un anneau de rayon R et d'axe (Oz) est chargé uniformément avec une densité linéique λ .



En un point M situé sur l'axe (Oz), le potentiel élémentaire est $dV(M) = \frac{k\lambda R d\theta}{PM}$; (P : point quelconque de l'anneau). On peut donc affirmer que le potentiel total créé par l'anneau au point M s'écrit :

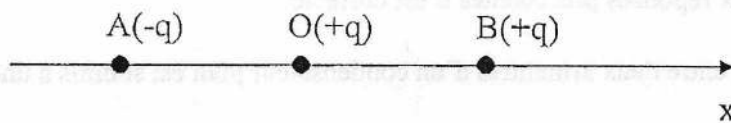
a) $V(z) = \frac{k\lambda R \pi}{\sqrt{z^2 + R^2}}$

b) $V(z) = \frac{2k\lambda R \pi z}{\sqrt{z^2 + R^2}}$

c) $V(z) = \frac{2k\lambda R \pi}{z^2 + R^2}$

d) $V(z) = \frac{2k\lambda R \pi}{\sqrt{z^2 + R^2}}$

50- Soit la distribution de charges représentée sur la figure ci-dessous : (AB = 2a et O est milieu de AB).



Le champ électrique créé au point A s'exprime par :

a) $E(A) = \frac{2k.q}{a^2}$

b) $E(A) = \frac{5k.q}{4a^2}$

c) $E(A) = \frac{3k.q}{4a^2}$

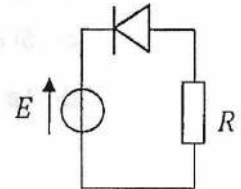
d) $E(A) = 0$

QCM Electronique – InfoS3

Pensez à bien lire les questions ET les réponses proposées (attention à la numérotation des réponses)

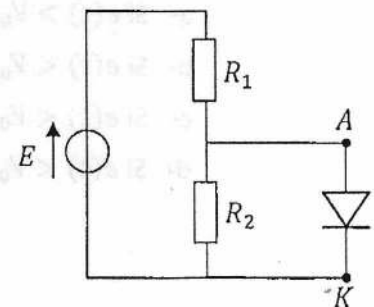
Q1. Soit le circuit ci-contre, dans lequel on considère la diode idéale. Que vaut la tension V_{AK} aux bornes de la diode si $E = 10V$, $R = 100\Omega$.

- a- $10V$
- b- $0V$
- c- $-10V$
- d- $0,7V$



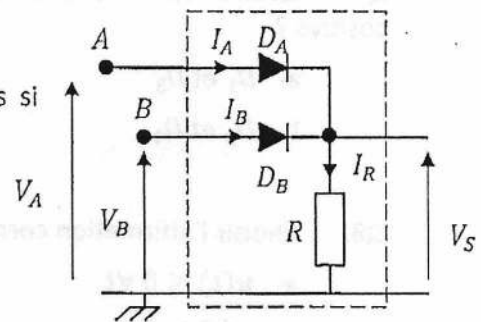
Q2. Soit le circuit ci-contre, dans lequel on modélise la diode par son modèle à seuil avec $V_0 = 0,6V$. Choisir l'affirmation correcte si $E = 1V$, $R_1 = 100\Omega$, et $R_2 = 50\Omega$:

- a- La diode est passante et le courant qui la traverse vaut $100mA$
- b- La diode est bloquée et la tension à ses bornes est égale à $\frac{1}{3}V$.
- c- La diode est passante et le courant qui la traverse vaut $5A$.
- d- La diode est passante et le courant qui la traverse vaut $200mA$.



Q3. Soit le circuit ci-contre : Comment sont les diodes si $V_A = V_B = 0V$? On supposera les diodes idéales.

- a- Bloquées
- b- Passantes



Q4. Si on veut montrer qu'une diode est passante par un raisonnement par l'absurde, il faut :

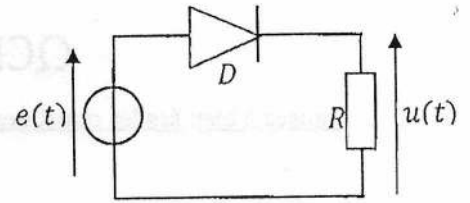
- a- La supposer bloquée et montrer que la tension à ses bornes est supérieure à sa tension de seuil.
- b- La supposer passante et montrer que la tension à ses bornes est supérieure à sa tension de seuil.
- c- La supposer passante et montrer que le courant qui la traverse de l'anode vers la cathode est positif.
- d- La supposer passante et montrer que le courant qui la traverse de l'anode vers la cathode est négatif.

11

Soit le circuit ci-contre, où $e(t) = E \cdot \sqrt{2} \cdot \sin(\omega \cdot t)$. (Q5&6)

Q5. On considère la diode idéale. Choisir les affirmations correctes :

- a- Si $e(t) > 0$, alors $u(t) = e(t)$.
- b- Si $e(t) < 0$, alors $u(t) = 0$.
- c- Si $e(t) > 0$, alors $u(t) = 0$.
- d- La diode est bloquée et la tension à ses bornes est égale à $\frac{E_0}{R} V$.



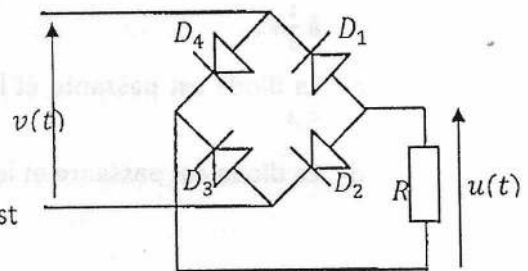
Q6. On utilise maintenant le modèle à seuil. Choisir l'affirmation correcte :

- a- Si $e(t) > V_0$, alors la diode est passante et la tension à ses bornes vaut 0.
- b- Si $e(t) < V_0$, alors la diode est passante et la tension à ses bornes vaut $e(t)$.
- c- Si $e(t) < V_0$, alors la diode est bloquée et la tension à ses bornes vaut V_0 .
- d- Si $e(t) < V_0$, alors la diode est bloquée et la tension à ses bornes vaut $e(t)$.

Soit le circuit suivant où $v(t) = V \cdot \sqrt{2} \cdot \sin(\omega t)$. (Q7 à Q9)

Q7. Quelles sont les diodes passantes si $v(t)$ est positive ?

- a- D_1 et D_3
- b- D_2 et D_4
- c- D_3 et D_4
- d- D_1 et D_2



Q8. Choisir l'affirmation correcte :

- a- $u(t) \leq 0 \forall t$
- b- $u(t) \geq 0 \forall t$
- c- $u(t) = 0$ si $v(t) \leq 0$
- d- $u(t) = 0$ si $v(t) \geq 0$

Q9. Que se passe-t-il si on modélise les diodes par leur modèle à seuil? On notera V_0 , la tension de seuil des diodes.

- a- Si $|v| > 2 \cdot V_0$, alors les 4 diodes sont bloquées.
- b- Si $|v| > V_0$, alors les 2 diodes de la question 7 sont passantes.
- c- Si $|v| < 2 \cdot V_0$, alors les 4 diodes sont bloquées.
- d- Toutes les réponses précédentes sont fausses.

Q10. Que se passe-t-il quand la tension appliquée aux bornes d'une diode devient très fortement négative (inférieure à une valeur spécifiée par le fabricant)

- a- Il ne se passe rien
- b- Le courant croît rapidement
- c- Le courant décroît rapidement et il peut y avoir destruction de la diode.
- d- Le courant croît puis devient nul.

QCM 4

Architecture des ordinateurs

Lundi 25 octobre 2021

Pour toutes les questions, une ou plusieurs réponses sont possibles.

11. À quoi sert le symbole '#' ?
 - A. Il indique qu'un opérande est une donnée immédiate.
 - B. Il indique qu'un opérande est sous forme décimale.
 - C. Il indique qu'un opérande est une adresse.
 - D. Il indique qu'un opérande est sous forme hexadécimale.
12. Quels modes d'adressage ne spécifient pas d'emplacement mémoire ?
 - A. Mode d'adressage immédiat.
 - B. Mode d'adressage direct.
 - C. Mode d'adressage absolu.
 - D. Mode d'adressage indirect.
13. Quelle(s) instruction(s) peut-on utiliser pour appeler un sous-programme ?
 - A. Aucune de ces réponses.
 - B. BEQ
 - C. JMP
 - D. BRA
14. Après l'exécution d'une instruction RTS, le pointeur de pile :
 - A. Aucune de ces réponses.
 - B. Ne change pas.
 - C. Est incrémenté de quatre.
 - D. Est décrémenté de quatre.
15. Les étapes pour empiler une donnée sont :
 - A. Incrémenter A7 puis écrire la donnée dans (A7).
 - B. Aucune de ces réponses.
 - C. Écrire la donnée dans (A7) puis décrémenter A7.
 - D. Décrémenter A7 puis écrire la donnée dans (A7).

16. L'instruction BCC effectue un branchement si :

- A. $C = 1$
- B. $V = 1$
- C. $C = 0$
- D. $V = 0$

17. L'instruction BVS effectue un branchement si :

- A. $C = 1$
- B. $V = 1$
- C. $C = 0$
- D. $V = 0$

18. Soient les deux instructions suivantes :

TST.W D0
BPL NEXT

L'instruction BPL effectue le branchement si :

- A. $D0 = \$88777788$
- B. $D0 = \$000000FF$
- C. $D0 = \$FFFFFF11$
- D. $D0 = \$FFFFF111$

19. Soient les deux instructions suivantes :

CMP.L D1,D2
BHI NEXT

L'instruction BHI effectue le branchement si :

- A. $D2 > D1$ (comparaison signée)
- B. $D1 > D2$ (comparaison signée)
- C. $D2 > D1$ (comparaison non signée)
- D. $D1 > D2$ (comparaison non signée)

20. Soient les deux instructions suivantes :

CMP.B D1,D2
BLT NEXT

Si $D2 = \$000000FF$, l'instruction BLT effectue le branchement si :

- A. $D1 = \$00000001$
- B. $D1 = \$000000FE$
- C. $D1 = \$FFFFFF01$
- D. Aucune de ces réponses.

15

EASy68K Quick Reference v1.8

<http://www.wowgweb.com/EASy68K.htm>

Copyright © 2004-2007 By: Chuck Kelly

Opcode	Size	Operand	CCR	Effective Address	s=source, d=destination, e=either, i=displacement	Operation	Description
	BWL	s,d	XN2VC	Dn An (An) (An)+ -(An) (An) (An,Rn) abs.W abs.L (LPC) (LPC,Rn) #n			
ABCD	B	Dy,Dx -(Ay),-(Ax)	*U*U*	e - - - - - - - - - -	e - - - - - - - - - -	$Dy_0 + Dx_{10} + X \rightarrow D_{x0}$ $-(Ay)_0 + -(Ax)_{10} + X \rightarrow -(Ax)_{10}$	Add BCD source and extend bit to destination. BCD result
ADD	BWL	s,Dn Dn,d	*****	e s s s s s s s s s s e d d d d d d d d d	e s s s s s s s s s s - - - - -	$s + Dn \rightarrow Dn$ $Dn + d \rightarrow d$	Add binary (ADD or ADDQ is used when source is #n. Prevent ADDQ with #n.L)
ADDA	WL	s,An	-----	s a s s s s s s s s s s	s a s s s s s s s s s s	$s + An \rightarrow An$	Add address (W sign-extended to L)
ADDI	BWL	#n,d	*****	d - d d d d d d d d d d	d - d d d d d d d d d d	$\#n + d \rightarrow d$	Add immediate to destination
ADDQ	BWL	#n,d	*****	d d d d d d d d d d	d d d d d d d d d d	$\#n + d \rightarrow d$	Add quick immediate (#n range: 1 to 8)
ADDX	BWL	Dy,Dx -(Ay),-(Ax)	*****	e - - - - - - - - - -	e - - - - - - - - - -	$Dy + Dx + X \rightarrow Dx$ $-(Ay) + -(Ax) + X \rightarrow -(Ax)$	Add source and extend bit to destination
AND	BWL	s,Dn Dn,d	---*00	e - s s s s s s s s s s e - d d d d d d d d d d	e - s s s s s s s s s s - - - - -	$s \text{ AND } Dn \rightarrow Dn$ $Dn \text{ AND } d \rightarrow d$	Logical AND source to destination (ANDI is used when source is #n)
ANDI	BWL	#n,d	---*00	d - d d d d d d d d d d	d - d d d d d d d d d d	$\#n \text{ AND } d \rightarrow d$	Logical AND immediate to destination
ANDI	B	#n,CCR	00 00 00 00	- - - - -	- - - - -	$\#n \text{ AND } CCR \rightarrow CCR$	Logical AND immediate to CCR
ANDI	W	#n,SR	00 00 00 00	- - - - -	- - - - -	$\#n \text{ AND } SR \rightarrow SR$	Logical AND immediate to SR (Privileged)
ASL	BWL	Dx,Dy	*****	e - - - - - d - - - - -	e - - - - - d - - - - -	$s \ll 1$ $Dx \ll 1$	Arithmetic shift Dy by Dx bits left/right
ASR	W	#n,Dy		- - d d d d d d d d d d	- - d d d d d d d d d d	$Dy \gg \#n$	Arithmetic shift Dy #n bits L/R (#n: 1 to 8)
Bcc	BW	address	-----	- - - - -	- - - - -	if cc true then address \rightarrow PC	Branch conditionally (cc table on back) (8 or 16-bit \pm offset to address)
BCHG	B L	Dn,d #n,d	---*--	e' - d d d d d d d d d d d' - d d d d d d d d d d	e' - d d d d d d d d d d d' - d d d d d d d d d d	$\text{NOT}(\text{bit number of } d) \rightarrow Z$ $\text{NOT}(\text{bit } n \text{ of } d) \rightarrow \text{bit } n \text{ of } d$	Set Z with state of specified bit in d then invert the bit in d
BCLR	B L	Dn,d #n,d	---*--	e' - d d d d d d d d d d d' - d d d d d d d d d d	e' - d d d d d d d d d d d' - d d d d d d d d d d	$\text{NOT}(\text{bit number of } d) \rightarrow Z$ $0 \rightarrow \text{bit number of } d$	Set Z with state of specified bit in d then clear the bit in d
BRA	BW	address	-----	- - - - -	- - - - -	address \rightarrow PC	Branch always (8 or 16-bit \pm offset to addr)
BSET	B L	Dn,d #n,d	---*--	e' - d d d d d d d d d d d' - d d d d d d d d d d	e' - d d d d d d d d d d d' - d d d d d d d d d d	$\text{NOT}(\text{bit } n \text{ of } d) \rightarrow Z$ $1 \rightarrow \text{bit } n \text{ of } d$	Set Z with state of specified bit in d then set the bit in d
BSR	BW	address	-----	- - - - -	- - - - -	PC \rightarrow (SP); address \rightarrow PC	Branch to subroutine (8 or 16-bit \pm offset)
BTST	B L	Dn,d #n,d	---*--	e' - d d d d d d d d d d d' - d d d d d d d d d d	e' - d d d d d d d d d d d' - d d d d d d d d d d	$\text{NOT}(\text{bit } Dn \text{ of } d) \rightarrow Z$ $\text{NOT}(\text{bit } \#n \text{ of } d) \rightarrow Z$	Set Z with state of specified bit in d Leave the bit in d unchanged
CHK	W	s,Dn	---UUU	e - s s s s s s s s s s	e - s s s s s s s s s s	if $Dn < 0$ or $Dn > s$ then TRAP	Compare Dn with D and upper bound (s)
CLR	BWL	d	-0100	d - d d d d d d d d d d	d - d d d d d d d d d d	$0 \rightarrow d$	Clear destination to zero
CMPI	BWL	s,Dn	---UUU	e s s s s s s s s s s	e s s s s s s s s s s	set CCR with $Dn - s$	Compare Dn to source
CMPI	WL	s,An	---UUU	s e s s s s s s s s s s	s e s s s s s s s s s s	set CCR with $An - s$	Compare An to source
CMPI	BWL	#n,d	---UUU	d - d d d d d d d d d d	d - d d d d d d d d d d	set CCR with $d - \#n$	Compare destination to #n
CMPI	BWL	(Ay),-(Ax)	---UUU	- - - e - - - - -	- - - e - - - - -	set CCR with $(Ax) - (Ay)$	Compare (Ax) to (Ay); Increment Ax and Ay
DBcc	W	Dn,address	-----	- - - - -	- - - - -	if cc false then { $Dn - 1 \rightarrow Dn$ if $Dn < -1$ then addr \rightarrow PC }	Test condition, decrement and branch (16-bit \pm offset to address)
DIVS	W	s,Dn	---*00	e - s s s s s s s s s s	e - s s s s s s s s s s	$\pm 32\text{bit } Dn / \pm 16\text{bit } s \rightarrow \pm Dn$	$Dn = (16\text{-bit remainder}, 16\text{-bit quotient})$
DIVU	W	s,Dn	---*00	e - s s s s s s s s s s	e - s s s s s s s s s s	$32\text{bit } Dn / 16\text{bit } s \rightarrow Dn$	$Dn = (16\text{-bit remainder}, 16\text{-bit quotient})$
EDR	BWL	Dn,d	---*00	e - d d d d d d d d d d	e - d d d d d d d d d d	$Dn \text{ XOR } d \rightarrow d$	Logical exclusive OR Dn to destination
EDRI	BWL	#n,d	---*00	d - d d d d d d d d d d	d - d d d d d d d d d d	$\#n \text{ XOR } d \rightarrow d$	Logical exclusive OR #n to destination
EDRI	B	#n,CCR	00 00 00 00	- - - - -	- - - - -	$\#n \text{ XOR } CCR \rightarrow CCR$	Logical exclusive OR #n to CCR
EDRI	W	#n,SR	00 00 00 00	- - - - -	- - - - -	$\#n \text{ XOR } SR \rightarrow SR$	Logical exclusive OR #n to SR (Privileged)
EXG	L	Rx,Ry	-----	e a - - - - -	e a - - - - -	register \leftrightarrow register	Exchange registers (32-bit only)
EXT	WL	Dn	---*00	d - - - - -	d - - - - -	$Dn.B \rightarrow Dn.W$ $Dn.W \rightarrow Dn.L$	Sign extend (change B to W or W to L)
ILLEGAL			-----	- - - - -	- - - - -	PC \rightarrow (SSP); SR \rightarrow (SSP)	Generate Illegal Instruction exception
JMP		d	-----	- - d - - - -	- - d - - - -	$\uparrow d \rightarrow PC$	Jump to effective address of destination
JSR		d	-----	- - d - - - -	- - d - - - -	PC \rightarrow (SP); $\uparrow d \rightarrow PC$	push PC, jump to subroutine at address d
LEA	L	s,An	-----	- e s - - - -	- e s - - - -	$\uparrow s \rightarrow An$	Load effective address of s to An
LINK		An,#n	-----	- - - - -	- - - - -	$An \rightarrow$ (SP); SP \rightarrow An; SP $+ \#n \rightarrow SP$	Create local workspace on stack (negative n to allocate space)
LSL	BWL	Dx,Dy	---*00	e - - - - - d - - - - -	e - - - - - d - - - - -	$s \ll 1$ $Dx \ll 1$	Logical shift Dy, Dx bits left/right
LSR	W	#n,Dy		- - d d d d d d d d d d	- - d d d d d d d d d d	$Dy \gg \#n$	Logical shift Dy, #n bits L/R (#n: 1 to 8)
MOVE	BWL	s,d	---*00	e s' a e e e e e e e s s	e s' a e e e e e e e s s	$s \rightarrow d$	Move data from source to destination
MOVE	W	s,CCR	00 00 00 00	s - s s s s s s s s s s	s - s s s s s s s s s s	$s \rightarrow CCR$	Move source to Condition Code Register
MOVE	W	s,SR	00 00 00 00	s - s s s s s s s s s s	s - s s s s s s s s s s	$s \rightarrow SR$	Move source to Status Register (Privileged)
MOVE	W	SR,d	-----	d - d d d d d d d d d d	d - d d d d d d d d d d	$SR \rightarrow d$	Move Status Register to destination
MOVE	L	USP,An	-----	- d - - - - -	- d - - - - -	$USP \rightarrow An$	Move User Stack Pointer to An (Privileged)
	BWL	An,USP	-----	- s - - - - -	- s - - - - -	$An \rightarrow USP$	Move An to User Stack Pointer (Privileged)

Opcode	Size	Operand	CCR	Effective Address s=source, d=destination, e=either, l=displacement												Operation	Description
	BWL	s,d	XNZVC	Un	An	(An)	(An)+	-(An)	(iAn)	(iAn,Rn)	abs.W	abs.L	(iPC)	(iPC,Rn)	#n		
MOVEA ⁴	WL	sAn	-----	s	e	s	s	s	s	s	s	s	s	s	s	s → An	Move source to An (MOVE sAn use MOVEA)
MOVE ⁴	WL	Rn-Rn,d s,Rn-Rn	-----	-	-	d	-	d	d	d	d	d	-	-	-	Registers → d s → Registers	Move specified registers to/from memory (W source is sign-extended to L for Rn)
MOVEP	WL	Dn,(iAn) (iAn),Dn	-----	s	-	-	-	-	d	-	-	-	-	-	-	Dn → (iAn)...(i+2An)...(i+4A) (iAn) → Dn...(i+2An)...(i+4A)	Move Dn to/from alternate memory bytes (Access only even or odd addresses)
MOVEQ ⁴	L	#n,Dn	---*00	d	-	-	-	-	-	-	-	-	-	-	s	#n → Dn	Move sign extended 8-bit #n to Dn
MULS	W	s,Dn	---*00	e	-	s	s	s	s	s	s	s	s	s	s	s ±16bit s * ±16bit Dn → zDn	Multiply signed 16-bit; result: signed 32-bit
MULU	W	s,Dn	---*00	e	-	s	s	s	s	s	s	s	s	s	s	s 16bit s * 16bit Dn → Dn	Multiply unsg'd 16-bit; result: unsg'd 32-bit
NBCD	B	d	*U*U*	d	-	d	d	d	d	d	d	d	-	-	-	0-d0-X → d	Negate BCD with eXtend, BCD result
NEG	BWL	d	*****	d	-	d	d	d	d	d	d	d	-	-	-	0-d → d	Negate destination (2's complement)
NEGX	BWL	d	*****	d	-	d	d	d	d	d	d	d	-	-	-	0-d-X → d	Negate destination with eXtend
NOP			-----	-	-	-	-	-	-	-	-	-	-	-	-	None	No operation occurs
NOT	BWL	d	---*00	d	-	d	d	d	d	d	d	d	-	-	-	NOT(d) → d	Logical NOT destination (1's complement)
OR ⁴	BWL	s,Dn Dn,d	---*00	e	-	s	s	s	s	s	s	s	s	s	s	s OR Dn → Dn Dn OR d → d	Logical OR (ORI is used when source is #n)
ORI ⁴	BWL	#n,d	---*00	d	-	d	d	d	d	d	d	d	-	-	s	#n OR d → d	Logical OR #n to destination
ORI ⁴	B	#n,CCR	000000	-	-	-	-	-	-	-	-	-	-	-	s	#n OR CCR → CCR	Logical OR #n to CCR
ORI ⁴	W	#n,SR	000000	-	-	-	-	-	-	-	-	-	-	-	s	#n OR SR → SR	Logical OR #n to SR (Privileged)
PEA	L	s	-----	-	-	s	-	-	s	s	s	s	s	s	-	Ts → (SP)	Push effective address of s onto stack
RESET			-----	-	-	-	-	-	-	-	-	-	-	-	-	Assert RESET Line	Issue a hardware RESET (Privileged)
ROL	BWL	Dx,Dy #n,Dy	---*0*	e	-	-	-	-	-	-	-	-	-	-	s		Rotate Dy, Dx bits left/right (without X)
ROR	W	d	---	d	-	d	d	d	d	d	d	d	-	-	-		Rotate Dy, #n bits left/right (#n: 1 to 8)
																	Rotate d l-bit left/right (W only)
ROXL	BWL	Dx,Dy #n,Dy	---*0*	e	-	-	-	-	-	-	-	-	-	-	s		Rotate Dy, Dx bits L/R, X used then updated
RDXR	W	d	---	d	-	d	d	d	d	d	d	d	-	-	-		Rotate Dy, #n bits left/right (#n: 1 to 8)
																	Rotate destination l-bit left/right (W only)
RTI			000000	-	-	-	-	-	-	-	-	-	-	-	-	(SP)+ → SR; (SP)+ → PC	Return from exception (Privileged)
RTR			000000	-	-	-	-	-	-	-	-	-	-	-	-	(SP)+ → CCR; (SP)+ → PC	Return from subroutine and restore CCR
RTS			-----	-	-	-	-	-	-	-	-	-	-	-	-	(SP)+ → PC	Return from subroutine
SBCD	B	Dy,Dx -(Ay),-(Ax)	*U*U*	e	-	-	-	-	-	-	-	-	-	-	-	Dx0-Dy10-X → Dx10 -(Ax)10- (Ay)10-X → -(Ax)10	Subtract BCD source and eXtend bit from destination, BCD result
SCC	B	d	-----	d	-	d	d	d	d	d	d	d	-	-	-	If cc is true then 1's → d else 0's → d	If cc true then d.B = 11111111 else d.B = 00000000
STOP		#n	000000	-	-	-	-	-	-	-	-	-	-	-	s	#n → SR; STOP	Move #n to SR, stop processor (Privileged)
SUB ⁴	BWL	s,Dn Dn,d	*****	e	s	s	s	s	s	s	s	s	s	s	s	Dn-s → Dn d-Dn → d	Subtract binary (SUBI or SUBQ used when source is #n. Prevent SUBQ with #n.L)
SUBA ⁴	WL	sAn	-----	s	e	s	s	s	s	s	s	s	s	s	s	An-s → An	Subtract address (W sign-extended to L)
SUBI ⁴	BWL	#n,d	*****	d	-	d	d	d	d	d	d	d	-	-	s	d-#n → d	Subtract immediate from destination
SUBQ ⁴	BWL	#n,d	*****	d	d	d	d	d	d	d	d	d	-	-	s	d-#n → d	Subtract quick immediate (#n range: 1 to 8)
SUBX	BWL	Dy,Dx -(Ay),-(Ax)	*****	e	-	-	-	-	-	-	-	-	-	-	-	Dx-Dy-X → Dx -(Ax)-(Ay)-X → -(Ax)	Subtract source and eXtend bit from destination
SWAP	W	Dn	---*00	d	-	-	-	-	-	-	-	-	-	-	-	bits(31:16) ↔ bits(15:0)	Exchange the 16-bit halves of Dn
TAS	B	d	---*00	d	-	d	d	d	d	d	d	d	-	-	-	test d → CCR: 1 → bit7 of d	N and Z set to reflect d, bit7 of d set to 1
TRAP		#n	-----	-	-	-	-	-	-	-	-	-	-	-	s	PC → (SSP); SR → (SSP); (vector table entry) → PC	Push PC and SR, PC set by vector table #n (#n range: 0 to 15)
TRAPV			-----	-	-	-	-	-	-	-	-	-	-	-	-	If V then TRAP #7	If overflow, execute an Overflow TRAP
TST	BWL	d	---*00	d	-	d	d	d	d	d	d	d	-	-	-	test d → CCR	N and Z set to reflect destination
UNLK		An	-----	-	d	-	-	-	-	-	-	-	-	-	-	An → SP; (SP)+ → An	Remove local workspace from stack
	BWL	s,d	XNZVC	Un	An	(An)	(An)+	-(An)	(iAn)	(iAn,Rn)	abs.W	abs.L	(iPC)	(iPC,Rn)	#n		

Condition Tests (* OR, I NOT, * XOR; * Unsigned, * Alternate cc)					
cc	Condition	Test	cc	Condition	Test
I	true	I	VC	overflow clear	IV
F	false	O	VS	overflow set	V
HI*	higher than	I(C+Z)	PL	plus	IN
LS*	lower or same	C+Z	MI	minus	N
HS*, CC*	higher or same	IC	GE	greater or equal	I(N ⊕ V)
LO*, CS*	lower than	C	LT	less than	(N ⊕ V)
NE	not equal	IZ	GT	greater than	I((N ⊕ V) + Z)
EQ	equal	Z	LE	less or equal	(N ⊕ V) + Z

Revised by Peter Csaszar, Lawrence Tech University – 2004-2006

Distributed under the GNU general public use license.

An Address register (16/32-bit, n=0-7)
 Dn Data register (8/16/32-bit, n=0-7)
 Rn any data or address register
 s Source, d Destination
 e Either source or destination
 #n Immediate data, l Displacement
 BCD Binary Coded Decimal
 † Effective address
 ‡ Long only; all others are byte only
 2 Assembler calculates offset
 3 Branch sizes: B or .S -128 to +127 bytes, .W or .L -32768 to +32767 bytes
 4 Assembler automatically uses A, L, Q or M form if possible. Use #n.L to prevent Quick optimization

SSP Supervisor Stack Pointer (32-bit)
 USP User Stack Pointer (32-bit)
 SP Active Stack Pointer (same as A7)
 PC Program Counter (24-bit)

SR Status Register (16-bit)
 CCR Condition Code Register (lower 8-bits of SR)
 N negative, Z zero, V overflow, C carry, X extend
 * set according to operation's result, = set directly
 - not affected, 0 cleared, 1 set, U undefined