Algo
QCM

1. **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,..,y)
   (d) Il existe un chemin (y,..,x)

2. **L'ordre d'un graphe orienté est ?**
   (a) Le nombre d'arcs du graphe
   (b) Le nombre de sommets du graphe
   (c) Le coût du graphe
   (d) La liste triée des arcs du graphe

3. **Un graphe orienté G défini par le triplet G=<S,A,C> est ?**
   (a) etiqueté
   (b) valué
   (c) valorisé
   (d) numéroté

4. **Un graphe peut être ?**
   (a) Orienté
   (b) Non orienté
   (c) A moitié orienté
   (d) Désorienté

5. **Dans un graphe orienté, on dit que l'arc $U = y \rightarrow x$ est ?**
   (a) incident à x vers l'extérieur
   (b) accident à x vers l'extérieur
   (c) incident à x vers l'intérieur
   (d) accident à x vers l'intérieur

6. **Dans un graphe orienté, le nombre d'arcs ayant le sommet x pour extrémité terminale est appelé ?**
   (a) le demi-degré extérieur de x
   (b) le degré de x
   (c) le demi-degré intérieur de x

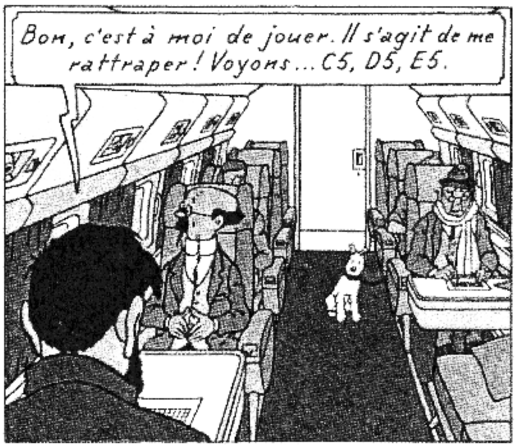
1

7. **Dans un graphe orienté, s'il existe un arc $U = y \rightarrow x$ pour tout couple de sommet $\{x, y\}$ le graphe est ?**
   (a) complet
   (b) partiel
   (c) parfait

8. **Dans un graphe orienté, un sommet de degré zéro est appelé ?**
   (a) sommet unique
   (b) sommet isolé
   (c) sommet nul
   (d) sommet perdu

9. **Deux arcs d'un graphe orienté sont dits adjacents si ?**
   (a) il existe deux arcs les joignant
   (b) le graphe est complet
   (c) ils ont au moins une extrémité commune

10. **Dans un graphe orienté valué G=<S,A,C>, les coûts sont portés par ?**
   (a) les arcs
   (b) les sommets



2

QCM N°5

lundi 19 novembre 2018

Question 11

Soit $A = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 3 \\ 1 & 1 & 2 \end{pmatrix}$. Alors le polynôme caractéristique de A est

a. $(1 - X)(X + 1)(X - 3)$

b. $(1 - X)(2 - X)$

c. $(1 - X)(2 - X)(X - 3)$

d. $(1 - X)^2(3 - X)$

e. rien de ce qui précède

Question 12

Soit $A = \begin{pmatrix} 3 & -1 & 1 \\ 0 & 2 & 0 \\ 1 & -1 & 3 \end{pmatrix}$. Alors les valeurs propres de A sont

a. 1 et 3
b. 1 et 2
c. 3 et 5
d. 1 et 4
e. rien de ce qui précède

Question 13

Soient E un $\mathbb{R}$-ev, $u \in \mathscr{L}(E)$, λ une valeur propre de u. Alors $x \in E_\lambda$ signifie

a. $u(\lambda x) = \lambda u(x)$

b. $u(x) = \lambda x$

c. $u(x) - \lambda x \neq 0$

d. $x \in \text{Im}(u - \lambda\, id)$

e. rien de ce qui précède

Question 14

Le déterminant $\begin{vmatrix} 1 & 0 & 1 \\ 2 & 1 & 3 \\ 1 & -1 & 1 \end{vmatrix}$ est égal à

a. -2
b. 2
c. 1
d. 3
e. rien de ce qui précède

3

Question 15

Soit $A \in \mathscr{M}_3(\mathbb{R})$. Alors A diagonalisable (dans $\mathbb{R}$) si et seulement si

a. P_A scindé dans $\mathbb{R}$ et $\forall \lambda \in \mathrm{Sp}_{\mathbb{R}}(A),\ \dim(E_\lambda) = m(\lambda)$

b. P_A scindé dans $\mathbb{R}$ et $\forall \lambda \in \mathrm{Sp}_{\mathbb{R}}(A),\ \dim(E_\lambda) \leqslant m(\lambda)$

c. P_A scindé dans $\mathbb{R}$ et $\forall \lambda \in \mathrm{Sp}_{\mathbb{R}}(A),\ \dim(E_\lambda) \neq m(\lambda)$

d. rien de ce qui précède

Question 16

a. L'application $\left\{ \begin{array}{ccc} \mathbb{R}^2 & \longrightarrow & \mathbb{R} \\ (x,y) & \longmapsto & |x+y| \end{array} \right.$ est linéaire

b. L'application $\left\{ \begin{array}{ccc} \mathbb{R}^2 & \longrightarrow & \mathbb{R} \\ (x,y) & \longmapsto & x+y \end{array} \right.$ est linéaire

c. L'application $\left\{ \begin{array}{ccc} \mathbb{R}^3 & \longrightarrow & \mathbb{R}^2 \\ (x,y,z) & \longmapsto & (x, y+z) \end{array} \right.$ est linéaire

d. rien de ce qui précède

Question 17

Soient E un $\mathbb{R}$-ev et $f \in \mathscr{L}(E)$. Alors

a. $\mathrm{Im}(f) = \{y \in E,\ \exists x \in E,\ x = f(y)\}$

b. $\mathrm{Ker}(f) = \{x \in E,\ f(x) = 0\}$

c. $\mathrm{Ker}(f) \subset \mathrm{Im}(f)$

d. $\mathrm{Im}(f) \subset \mathrm{Ker}(f)$

e. rien de ce qui précède

Question 18

Soient E un $\mathbb{R}$-ev et $(f,g) \in (\mathscr{L}(E))^2$ quelconque. Alors

a. $\mathrm{Ker}(g) \subset \mathrm{Ker}(g \circ f)$

b. $\mathrm{Ker}(g \circ f) \subset \mathrm{Ker}(f)$

c. $\mathrm{Im}(f) \subset \mathrm{Im}(g \circ f)$

d. $\mathrm{Im}(g \circ f) \subset \mathrm{Im}(f)$

e. rien de ce qui précède

4

Question 19

Soient $E = \mathbb{R}_2[X]$ et $F = \text{Vect}\big(\{1, 1+X, 1-X^2\}\big)$. Alors

a. La famille $\big(1, 1+X, 1-X^2\big)$ est libre

b. F est un sev de E

c. $\dim(F) = \dim(E)$

d. $F = E$

e. rien de ce qui précède

Question 20

Soient E un $\mathbb{R}$-ev et $f \in \mathscr{L}(E)$. Alors f injective ssi

a. $\text{Ker}(f) = \{0\}$

b. $\forall (x,y) \in E^2 \ : \quad x = y \Rightarrow f(x) = f(y)$

c. $\forall (x,y) \in E^2 \ : \quad f(x) = f(y) \Rightarrow x = y$

d. rien de ce qui précède

QCM Azar Chap13 (Adjective clauses ex 8, 11, 13) QCM 5

In 21 - 24, the two sentences have been combined. Which of the combinations use the second sentence as an adjective clause correctly? (Punctuation is taken into account.)

21. The book was good. I read it.
a. The book was good that I read.
b. The book that I read was good.
c. The book I read it was good.
d. B and C.

22. I liked the woman. I met her at the party last night.
a. I liked the woman, that I met her at the party last night.
b. The woman I liked I met at the party last night.
c. I met at the party last night the woman that I liked.
d. I met at the party last night the woman whom I liked.
e. I liked the woman that I met at the party last night.

23. I liked the song. My brother wrote it.
a. I liked the song that my brother wrote it.
b. I liked the song that my brother wrote.
c. My brother wrote the song I liked.
d. I liked the song, that my brother wrote it.

24. The people were very nice. We visited them yesterday.
a. The people, we visited them yesterday, were very nice
b. We visited the people whom were very nice yesterday.
c. The people whom we visited yesterday were very nice.
d. The people we visited yesterday were very nice.
e. C and D.

Choose the adjective clause that is **NOT** correct for the following sentences.

25. The keys ___ were under the table.
a. that I was looking for
b. I was looking for
c. which I was looking for
d. whom I was looking for

26. The man ___ at the health care center was able to answer most of my questions.
a. who I spoke to
b. to who I spoke
c. to whom I spoke
d. I spoke to
e. All of the above.

Identify the adjective clause in the following sentences.

27. I returned the money which I had borrowed from my parents.
a. I returned the money
b. which I had borrowed from my parents
c. from my parents
d. A and B

28. Yesterday on the bus I ran into a man I had shared a room with at college.

a. on the bus
b. I ran into a man
c. I had shared a room with at college
d. I ran into a man I had shared a room

29. Anne talked in detail about a movie that she did not see.

a. a movie she did not see
b. Anne talked in detail about a movie
c. a movie that she did not see
d. that she did not see

30. Did you read about the candidate who is accused of tax evasion?

a. Did you read about
b. the candidate who is accused
c. who is accused of tax evasion
d. None of the above.

31) The study is being carried out in _________.
a) Columbia
b) Brazil
c) France
d) USA

32) The article is investigating __________________.
a) what people wear
b) people's behavior
c) job profiles
d) None of the above

33) The study used facial recognition to _________ images which had people's faces.
a) avoid
b) reveal
c) blur
d) None of the above

34) They were looking for __________________ themes from one location to another.
a) spikes in
b) similarities on
c) differences between
d) family

35) Red is showing to be the color that people are _________.
a) wearing less
b) increasingly wearing
c) wearing constantly
d) None of the above

36) Some countries follow the colors reflecting their national ___________________.
a) sports teams
b) filmstars
a) flags
b) None of the above

37) The study __________ a representation of society as a whole.
a) is not
b) is
c) will be
d) None of the above

38) The shortcomings of the project are that it:
a) is mainly showing younger people
b) only captures the upper part of the body
c) Neither of the above
d) Both of the above

39) The visual data can be used with other data like ____________________.
a) meteorological
b) music listening trends
c) eating habits
d) All of the above

40) This kind of ‘automated’ study shows that the job of an anthropologist may be ______________ than before.

a) more necessary
b) less necessary
c) as necessary
d) None of the above

The algorithm reveals how clothing styles change over time

Data-Mining 100 Million Instagram Photos Reveals Global Clothing Patterns

The millions of photos uploaded to social media are a massive untapped resource for studying humanity. But machine learning is beginning to tap this mother lode.

by Emerging Technology from the arXiv June 15, 2017

"Imagine a future anthropologist with access to trillions of photos of people—taken over centuries and across the world—and equipped with effective tools for analyzing these photos to derive insights. What kinds of new questions can be answered?"

This is the dream that has inspired Kevin Matzen, Kavita Bala, and Noah Snavely at Cornell University in Ithaca, New York.

Their thinking is that the millions of photos uploaded each day to social media provide a fascinating window into the cultural, social, and economic factors that shape societies around the world. With powerful enough machine intelligence, they say, it ought to be possible to mine this mother lode of data for deep insights into our civilization.

As luck would have it, this kind of machine intelligence is currently emerging at breakneck speed. And Matzen and co have put it to work studying 100 million photos posted on Instagram. The question these guys specifically want to answer was how clothing styles vary around the world, a cultural phenomenon that is otherwise difficult to study on this scale. For example, their approach can tackle questions such as how the frequency of scarf use in the U.S. is changing over time, what styles are most specific to particular regions or cities, and, conversely, which styles are popular across the world.

To find out, Matzen and co turned to Instagram, which allowed them to download images within five kilometers of a specific location and within five days of a specific date. The team then identified 44 cities to study and downloaded a total of 100 million images from these locations in five-day windows between June 2013 and June 2016.

How they did it...

They used a standard face recognition program to filter out all the pictures that did not contain a face, and they also filtered for a visible torso, leaving a set of 15 million photos of people showing the upper half of their body, along with their location and the date.

Next, they trained a machine-learning algorithm to recognize various types of clothing and accessories in images. For example, the algorithm learned to recognize whether people were wearing a jacket, a scarf, a necktie, glasses, a hat, and so on. The algorithm could also recognize colors, neckline styles, and sleeve length; clothing categories such as T-shirt, dress, or tank top; and clothing patterns, such as solid, striped, plaid, and so on.

Finally, they let the machine loose on the 15 million photos in their data set and then used another algorithm to search for clusters of images with similar visual themes and track how these varied across time and from one location to another.

The results make for interesting reading. The clustering algorithm found some 400 different visual themes, such as people wearing white T-shirts and glasses, or wearing red V-neck tops or black dresses, or not wearing tops at all!

Matzen and co can then study how these visual themes vary by time and place. They found, for example, that certain colors vary periodically, with black and brown being more common in winter and white and blue more common in summer.

Other colors show different patterns. For example, the popularity of red is dropping. And although it is much less periodic than black or white, it does become suddenly popular from time to time.

Matzen and co point to small spikes in popularity near the end of October and December: in other words, at Halloween and Christmas.

"What stood out were a large assortment of Santa hats as well as an unexpected assortment of red Halloween costumes with red hats or hoods," they say.

They also found a sudden increase in popularity of yellow shirts in Colombia and Brazil during the June/July 2014 football World Cup—both countries' football teams wear yellow. They also noted various geographical trends. "Countries further north tend to feature more jackets," they say, presumably because it is colder.

Hat wearing is also more popular in colder countries. But curiously, Oman in the Middle East turns out to be one of the hat-wearing capitals of the world. "In particular, the kuma and massar are popular in Oman, as they are an important element of the men's national dress," say Matzen and co.

Some clothes are unique to particular places: the gele, a Nigerian head-tie, is very distinctive of Lagos. And other styles are common around the world and throughout the year, including blue collared shirts, plaid shirts, and black T-shirts.

That's interesting work that reveals the potential for machine learning to tease apart the cultural fabric of our society.

Of course, this approach is not perfect. The algorithm did not learn to distinguish between sunglasses and prescription glasses, which play different roles in society. The images are unlikely to be representative of society as a whole, since Instagram users are heavily skewed toward a younger demographic. And the technique only looks at the upper body, since the legs are often cropped in online images.

But there is significant potential to correct these shortcomings in future work and to go further. An ongoing challenge in machine vision is to work out whether people are standing or sitting or what they are doing in general. It would also be possible to combine this data set with others, such as weather and temperature data.

As Matzen and co conclude: "The combination of big data, machine learning, computer vision, and automated analysis algorithms would make for a very powerful analysis tool more broadly in visual discovery of fashion and many other areas."

Clearly, we don't need to wait for the anthropologists of the future.

Ref: arxiv.org/abs/1706.01869 : StreetStyle: Exploring World-Wide Clothing Styles from Millions of Photos. REF: https://www.technologyreview.com/s/608116/data-mining-100-million-instagram-photos-reveals-global-clothing-patterns/?utm_content=buffer6aab8&utm_medium=social&utm_source=facebook.com&utm_campaign=buffer

EPITA-S3
2018/2019

Q.C.M n°5 de Physique

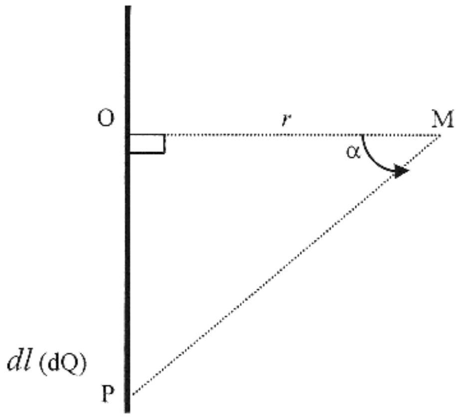
41- Soit la fonction $f(r) = r.e^{\frac{1}{r}}$. Son gradient peut s'écrire :

a) $\overrightarrow{grad}(f) = \left(e^{\frac{1}{r}} - \frac{1}{r}e^{\frac{1}{r}}\right).\overrightarrow{u_r}$
b) $\overrightarrow{grad}(f) = 2e^{\frac{1}{r}}.\overrightarrow{u_r}$
c) $\overrightarrow{grad}(f) = \left(e^{\frac{1}{r}} + \frac{1}{r}e^{\frac{1}{r}}\right).\overrightarrow{u_r}$

42- En coordonnées polaires (r, θ), quel élément infinitésimal $\overrightarrow{dl}$ de longueur existe ?

a) $\overrightarrow{dl} = d\theta.\overrightarrow{u_\theta}$ b) $\overrightarrow{dl} = dr.\overrightarrow{u_r}$ c) $\overrightarrow{dl} = rd\theta.\overrightarrow{u_r}$

43- On montre qu'un élément infinitésimal situé en P d'un fil de charge linéique λ crée un champ électrique en un point M extérieur au fil $dE_x(x) = \frac{k.\lambda}{x}\cos(\alpha)d\alpha$ où α est tel qu'indiqué ci-dessous.



Le champ électrique créé par un fil infini vaut :

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

44- Une distribution de charges sphérique crée au point M un potentiel électrique $V(\theta, \varphi)$. On peut affirmer que le vecteur champ électrique s'écrira :

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

45- Le champ électrique $\vec{E}(r) = k\frac{Qr}{a_0^3}e^{-\frac{r^2}{a_0^2}}.\overrightarrow{u_r}$, où Q et a_0 sont des constantes, dérive du potentiel :

a) $V(r) = k\frac{Q}{2a_0}e^{-\frac{r^2}{a_0^2}}$ b) $V(r) = -k\frac{Q}{a_0}e^{-\frac{r^2}{a_0^2}}$ c) $V(r) = k\frac{Q}{a_0^2}e^{-\frac{r^2}{a_0^2}}$

46- Le champ $\vec{E}(M)$, généré par un cylindre creux infini de rayon a chargé avec une densité surfacique σ uniforme, en un point M à l'intérieur du cylindre, est donné par :

a) $\vec{E}(M) = \vec{0}$ b) $\vec{E}(M) = \frac{k\sigma a}{r}\overrightarrow{u_r}$ c) $\vec{E}(M) = \frac{k\sigma}{r^2}\overrightarrow{u_r}$

47- On considère une distribution de charges invariante par translation selon (Ox) et (Oy). On note $\vec{E}$ le champ créé par cette distribution. Quelle propriété est vraie ?

a) $E_x = 0$ b) $\frac{\partial \vec{E}}{\partial z} = \vec{0}$ c) $\frac{\partial \vec{E}}{\partial x} = \vec{0}$

48- En reprenant la géométrie de la question 46, lors du calcul du champ $\vec{E}$, quelle surface de Gauss est la plus pertinente ?

a) Un cylindre ouvert b) Un cylindre fermé c) Une sphère

49- Pour toute surface S, le flux ϕ du champ $\vec{E}$ est donné par :

a) $\phi = \iint_S \frac{\partial \vec{E}}{\partial t}.\overrightarrow{dS}$ b) $\phi = \iint_S E.dS$ c) $\phi = \iint_S \vec{E}.\overrightarrow{dS}$

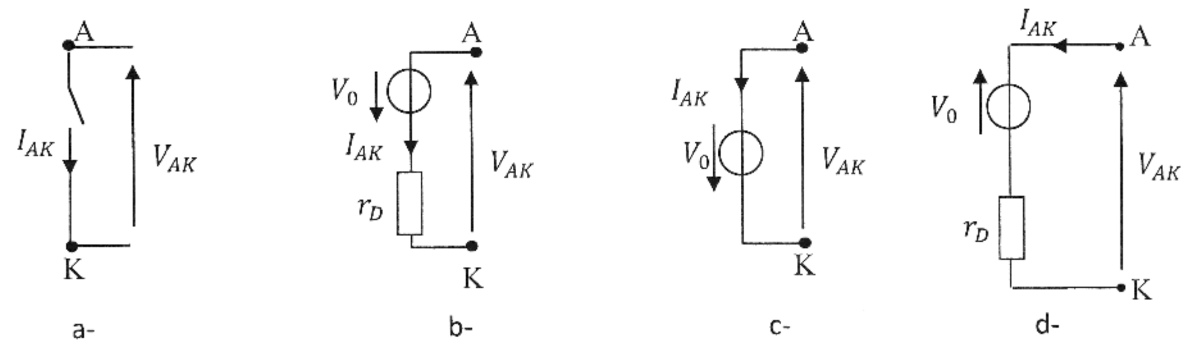
50- Que vaut alors le flux de $\vec{E}$ à travers un disque de rayon R ? On simplifiera en prenant un champ uniforme qui forme un angle α avec l'axe du disque.

a) $\phi = \pi R^2.\|\vec{E}\|.\cos(\alpha)$
b) $\phi = 4\pi R^2.\|\vec{E}\|.\cos(\alpha)$
c) $\phi = \pi R^2.\|\vec{E}\|$

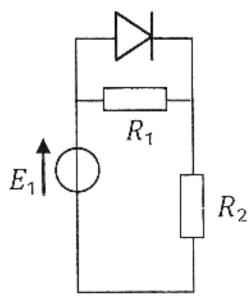
QCM Electronique – InfoS3

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

Q1. Par quoi remplace-t-on la diode passante si on utilise le modèle réel?



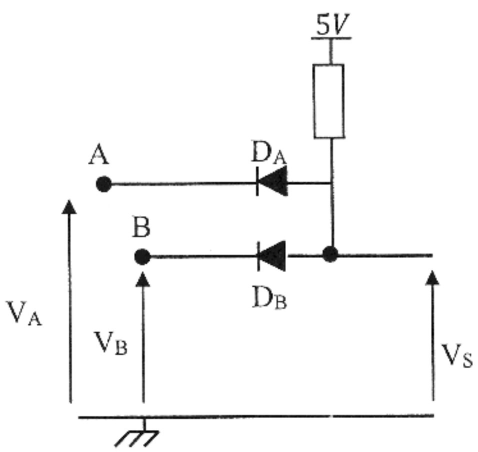
Q2. Soit le circuit ci-contre, dans lequel on modélise la diode par son modèle à seuil avec $V_0 = 0,6V$. Choisir l'affirmation correcte si $E_1 = 10V$, $R_1 = 50\Omega$, et $R_2 = 1k\Omega$:



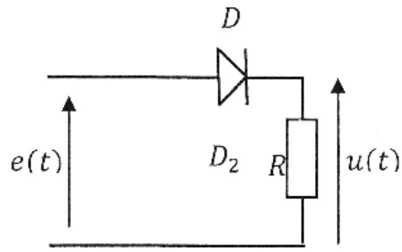
- a- La diode est bloquée et la tension à ses bornes est de l'ordre de 0,5V.
- b- La diode est passante et le courant qui la traverse est de l'ordre de $10\ mA$
- c- La diode est passante et le courant qui la traverse vaut $-5A$.
- d- La diode est passante et le courant qui la traverse est de l'ordre de $9,4\ mA$.

Q3. Soit le circuit ci-contre. Quel type de porte logique réalise ce montage ?

- a- OU
- b- NON ET
- c- ET
- d- NON OU



Soit le circuit ci-contre, où $e(t) = E_0.sin(\omega.t)$. (Q4&5)



Q4. On considère la diode idéale. Choisir l'affirmation correcte :

a- La diode est bloquée et la tension à ses bornes est égale à $\frac{E_0}{R}V$.

b- Si $e(t) < 0$, alors la diode est passante.

c- Si $e(t) < 0$, alors la diode est bloquée.

d- Si $e(t) > 0$, alors la diode est bloquée.

Q5. 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.

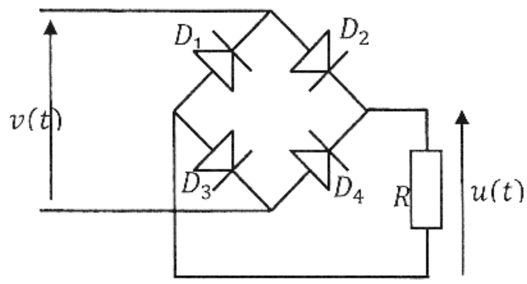
b- Si $e(t) < V_0$, alors la diode est passante et la tension à ses bornes vaut $e(t)$.

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. On supposera les diodes idéales. (Q6 & 7)

Q6. Quelles sont les diodes passantes si $v(t)$ est positive?



a- D_1 et D_4

b- D_1 et D_3

c- D_2 et D_4

d- D_2 et D_3

Q7. Quelle est l'expression de $u(t)$ quand $v(t)$ est négative ?

a- $u(t) = 0$

b- $u(t) = v(t)$

c- $u(t) = -v(t)$

d- $u(t) = v(t) - 2V_0$

Q8. En polarisation directe, la diode Zéner se comporte comme un générateur de courant.

a- VRAI

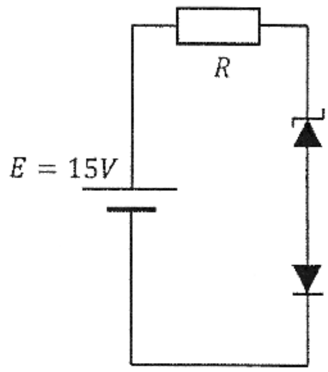
b- FAUX

Q9. En polarisation inverse, on peut représenter la diode Zéner à l'aide de l'un des 2 modèles : à seuil ou réel – le modèle idéal n'existant pas pour cette diode.

a- VRAI

b- FAUX

Q10. Soit le circuit ci-contre : On suppose que la tension de seuil inverse de la diode Zéner est de 6V. La diode Zéner est : (1 ou plusieurs réponses sont possibles.)



a- Polarisée en inverse

b- Passante

c- Polarisée en directe

d- Bloquée

QCM 5
Architecture des ordinateurs

Lundi 19 novembre 2018

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

11. Soient les deux instructions suivantes :

```
CMP.L D1,D2
BGT   NEXT
```

L'instruction BGT effectue le branchement si :

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

12. Soient les deux instructions suivantes :

```
CMP.L D1,D2
BLO   NEXT
```

L'instruction BLO effectue le branchement si :

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

13. Soient les cinq instructions suivantes :

```
MOVE.L  (A7)+,D2
MOVE.L  (A7)+,D3
MOVE.L  (A7)+,D4
MOVE.L  (A7)+,A4
MOVE.L  (A7)+,A5
```

Elles sont équivalentes à :

A. MOVEM.L (A7)+,D4/D2/D3/A4/A5
B. MOVEM.L (A7)+,A5/A4/D3/D2/D4
C. MOVEM.L (A7)+,D2-D4/A4/A5
D. MOVEM.L (A7)+,A5/A4-D3/D2/D4

14. Après l'exécution d'une instruction `RTS`, le pointeur de pile est :
    A. Décrémenté de quatre.
    B. Incrémenté de quatre.
    C. Incrémenté de deux.
    D. Décrémenté de deux.

15. Les étapes pour dépiler une donnée sont :
    A. Lire la donnée dans (A7) puis incrémenter A7.
    B. Incrémenter A7 puis lire la donnée dans (A7).
    C. Décrémenter A7 puis écrire la donnée dans (A7).
    D. Écrire la donnée dans (A7) puis décrémenter A7.

16. L'instruction `RTS` :
    A. Empile une adresse de retour.
    B. Est une instruction de saut.
    C. Ne modifie pas la pile.
    D. Restaure les registres.

17. Quelle(s) instruction(s) peut-on utiliser pour appeler un sous-programme ?
    A. BSR
    B. JSR
    C. ASR
    D. LSR

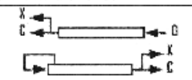
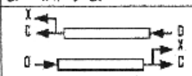
18. Quelle instruction n'est pas possible ?
    A. ADDI.L #1,D0
    B. ADDQ.L #8,D2
    C. ADDI.L #25,D1
    D. ADDQ.L #19,D3

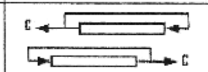
19. Quelles instructions ne sont pas possibles ?
    A. MULU.L #50,D0
    B. MULU.W #50,D0
    C. MULS.L #$50,D0
    D. MULS.W #$50,D0

20. Quelle opération arithmétique réalise l'instruction suivante ? `LSR.L #5,D0`
    A. D0 × 32
    B. D0 / 32
    C. D0 × 5
    D. D0 / 5

18

EASy68K Quick Reference v1.8 http://www.wowgwep.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	XNZVC	Dn	An	(An)	(An)+	-(An)	(i,An)	(i,An,Rn)	abs.W	abs.L	(i,PC)	(i,PC,Rn)	#n		
ABCD	B	Dy,Dx	*U*U*	e	-	-	-	-	-	-	-	-	-	-	-	$Dy_{10} + Dx_{10} + X \rightarrow Dx_{10}$	Add BCD source and eXtend bit to
		-(Ay),-(Ax)		-	-	-	-	e	-	-	-	-	-	-	-	$-(Ay)_{10} + -(Ax)_{10} + X \rightarrow -(Ax)_{10}$	destination, BCD result
ADD [4]	BWL	s,Dn	*****	e	s	s	s	s	s	s	s	s	s	s	s[4]	s + Dn → Dn	Add binary (ADDI or ADDQ is used when
		Dn,d		e	d[4]	d	d	d	d	d	d	d	-	-	-	Dn + d → d	source is #n. Prevent ADDQ with #n.L)
ADDA [4]	WL	s,An	-----	s	e	s	s	s	s	s	s	s	s	s	s	s + An → An	Add address (.W sign-extended to .L)
ADDI [4]	BWL	#n,d	*****	d	-	d	d	d	d	d	d	d	-	-	s	#n + d → d	Add immediate to destination
ADDQ [4]	BWL	#n,d	*****	d	d	d	d	d	d	d	d	d	-	-	s	#n + d → d	Add quick immediate (#n range: 1 to 8)
ADDX	BWL	Dy,Dx	*****	e	-	-	-	-	-	-	-	-	-	-	-	Dy + Dx + X → Dx	Add source and eXtend bit to destination
		-(Ay),-(Ax)		-	-	-	-	e	-	-	-	-	-	-	-	-(Ay) + -(Ax) + X → -(Ax)	
AND [4]	BWL	s,Dn	-**00	e	-	s	s	s	s	s	s	s	s	s	s[4]	s AND Dn → Dn	Logical AND source to destination
		Dn,d		e	-	d	d	d	d	d	d	d	-	-	-	Dn AND d → d	(ANDI is used when source is #n)
ANDI [4]	BWL	#n,d	-**00	d	-	d	d	d	d	d	d	d	-	-	s	#n AND d → d	Logical AND immediate to destination
ANDI [4]	B	#n,CCR	=====	-	-	-	-	-	-	-	-	-	-	-	s	#n AND CCR → CCR	Logical AND immediate to CCR
ANDI [4]	W	#n,SR	=====	-	-	-	-	-	-	-	-	-	-	-	s	#n AND SR → SR	Logical AND immediate to SR (Privileged)
ASL	BWL	Dx,Dy	*****	e	-	-	-	-	-	-	-	-	-	-	-		Arithmetic shift Dy by Dx bits left/right
ASR		#n,Dy		d	-	-	-	-	-	-	-	-	-	-	s		Arithmetic shift Dy #n bits L/R (#n: 1 to 8)
	W	d		-	-	d	d	d	d	d	d	d	-	-	-		Arithmetic shift ds 1 bit left/right (.W only)
Bcc	BW[3]	address[2]	-----	-	-	-	-	-	-	-	-	-	-	-	-	if cc true then	Branch conditionally (cc table on back)
																address → PC	(8 or 16-bit ± offset to address)
BCHG	B L	Dn,d	--*--	e[1]	-	d	d	d	d	d	d	d	-	-	-	NOT(bit number of d) → Z	Set Z with state of specified bit in d then
		#n,d		d[1]	-	d	d	d	d	d	d	d	-	-	s	NOT(bit n of d) → bit n of d	invert the bit in d
BCLR	B L	Dn,d	--*--	e[1]	-	d	d	d	d	d	d	d	-	-	-	NOT(bit number of d) → Z	Set Z with state of specified bit in d then
		#n,d		d[1]	-	d	d	d	d	d	d	d	-	-	s	0 → bit number of d	clear the bit in d
BRA	BW[3]	address[2]	-----	-	-	-	-	-	-	-	-	-	-	-	-	address → PC	Branch always (8 or 16-bit ± offset to addr)
BSET	B L	Dn,d	--*--	e[1]	-	d	d	d	d	d	d	d	-	-	-	NOT(bit n of d) → Z	Set Z with state of specified bit in d then
		#n,d		d[1]	-	d	d	d	d	d	d	d	-	-	s	1 → bit n of d	set the bit in d
BSR	BW[3]	address[2]	-----	-	-	-	-	-	-	-	-	-	-	-	-	PC → -(SP); address → PC	Branch to subroutine (8 or 16-bit ± offset)
BTST	B L	Dn,d	--*--	e[1]	-	d	d	d	d	d	d	d	d	d	-	NOT(bit Dn of d) → Z	Set Z with state of specified bit in d
		#n,d		d[1]	-	d	d	d	d	d	d	d	d	d	s	NOT(bit #n of d) → Z	Leave the bit in d unchanged
CHK	W	s,Dn	-*UUU	e	-	s	s	s	s	s	s	s	s	s	s	if Dn<0 or Dn>s then TRAP	Compare Dn with 0 and upper bound (s)
CLR	BWL	d	-0100	d	-	d	d	d	d	d	d	d	-	-	-	0 → d	Clear destination to zero
CMP [4]	BWL	s,Dn	-****	e	s[4]	s	s	s	s	s	s	s	s	s	s[4]	set CCR with Dn – s	Compare Dn to source
CMPA [4]	WL	s,An	-****	s	e	s	s	s	s	s	s	s	s	s	s	set CCR with An – s	Compare An to source
CMPI [4]	BWL	#n,d	-****	d	-	d	d	d	d	d	d	d	-	-	s	set CCR with d - #n	Compare destination to #n
CMPM [4]	BWL	(Ay)+,(Ax)+	-****	-	-	-	e	-	-	-	-	-	-	-	-	set CCR with (Ax) - (Ay)	Compare (Ax) to (Ay); Increment Ax and Ay
DBcc	W	Dn,addres[2]	-----	-	-	-	-	-	-	-	-	-	-	-	-	if cc false then { Dn-1 → Dn	Test condition, decrement and branch
																if Dn <> -1 then addr → PC }	(16-bit ± offset to address)
DIVS	W	s,Dn	-***0	e	-	s	s	s	s	s	s	s	s	s	s	±32bit Dn / ±16bit s → ±Dn	Dn= [16-bit remainder, 16-bit quotient]
DIVU	W	s,Dn	-***0	e	-	s	s	s	s	s	s	s	s	s	s	32bit Dn / 16bit s → Dn	Dn= [16-bit remainder, 16-bit quotient]
EOR [4]	BWL	Dn,d	-**00	e	-	d	d	d	d	d	d	d	-	-	s[4]	Dn XOR d → d	Logical exclusive OR Dn to destination
EORI [4]	BWL	#n,d	-**00	d	-	d	d	d	d	d	d	d	-	-	s	#n XOR d → d	Logical exclusive OR #n to destination
EORI [4]	B	#n,CCR	=====	-	-	-	-	-	-	-	-	-	-	-	s	#n XOR CCR → CCR	Logical exclusive OR #n to CCR
EORI [4]	W	#n,SR	=====	-	-	-	-	-	-	-	-	-	-	-	s	#n XOR SR → SR	Logical exclusive OR #n to SR (Privileged)
EXG	L	Rx,Ry	-----	e	e	-	-	-	-	-	-	-	-	-	-	register ←→ register	Exchange registers (32-bit only)
EXT	WL	Dn	-**00	d	-	-	-	-	-	-	-	-	-	-	-	Dn.B → Dn.W | Dn.W → Dn.L	Sign extend (change .B to .W or .W to .L)
ILLEGAL			-----	-	-	-	-	-	-	-	-	-	-	-	-	PC → -(SSP); SR → -(SSP)	Generate Illegal Instruction exception
JMP		d	-----	-	-	d	-	-	d	d	d	d	d	d	-	↑d → PC	Jump to effective address of destination
JSR		d	-----	-	-	d	-	-	d	d	d	d	d	d	-	PC → -(SP); ↑d → PC	push PC, jump to subroutine at address d
LEA	L	s,An	-----	-	e	s	-	-	s	s	s	s	s	s	-	↑s → An	Load effective address of s to An
LINK		An,#n	-----	-	-	-	-	-	-	-	-	-	-	-	-	An → -(SP); SP → An;	Create local workspace on stack
																SP + #n → SP	(negative n to allocate space)
LSL	BWL	Dx,Dy	***0*	e	-	-	-	-	-	-	-	-	-	-	-		Logical shift Dy, Dx bits left/right
LSR		#n,Dy		d	-	-	-	-	-	-	-	-	-	-	s		Logical shift Dy, #n bits L/R (#n: 1 to 8)
	W	d		-	-	d	d	d	d	d	d	d	-	-	-		Logical shift d 1 bit left/right (.W only)
MOVE [4]	BWL	s,d	-**00	e	s[4]	e	e	e	e	e	e	e	s	s	s[4]	s → d	Move data from source to destination
MOVE	W	s,CCR	=====	s	-	s	s	s	s	s	s	s	s	s	s	s → CCR	Move source to Condition Code Register
MOVE	W	s,SR	=====	s	-	s	s	s	s	s	s	s	s	s	s	s → SR	Move source to Status Register (Privileged)
MOVE	W	SR,d	-----	d	-	d	d	d	d	d	d	d	-	-	-	SR → d	Move Status Register to destination
MOVE	L	USP,An	-----	-	d	-	-	-	-	-	-	-	-	-	-	USP → An	Move User Stack Pointer to An (Privileged)
		An,USP		-	s	-	-	-	-	-	-	-	-	-	-	An → USP	Move An to User Stack Pointer (Privileged)
	BWL	s,d	XNZVC	Dn	An	(An)	(An)+	-(An)	(i,An)	(i,An,Rn)	abs.W	abs.L	(i,PC)	(i,PC,Rn)	#n		

Opcode	Size	Operand	CCR	Dn	An	(An)	(An)+	-(An)	(i,An)	(i,An,Rn)	abs.W	abs.L	(i,PC)	(i,PC,Rn)	#n	Operation	Description
	BWL	s,d	XNZVC	Effective Address s=source, d=destination, e=either, i=displacement													
MOVEA⁴	WL	s,An	-----	s	e	s	s	s	s	s	s	s	s	s	s	s → An	Move source to An (MOVE s,An use MOVEA)
MOVEM⁴	WL	Rn-Rn,d	-----	-	-	d	-	d	d	d	d	d	-	-	-	Registers → d	Move specified registers to/from memory
		s,Rn-Rn		-	-	s	s	-	s	s	s	s	s	s	-	s → Registers	(.W source is sign-extended to .L for Rn)
MOVEP	WL	Dn,(i,An)	-----	s	-	-	-	-	d	-	-	-	-	-	-	Dn → (i,An)...(i+2,An)...(i+4,A.	Move Dn to/from alternate memory bytes
		(i,An),Dn		d	-	-	-	-	s	-	-	-	-	-	-	(i,An) → Dn...(i+2,An)...(i+4,A.	(Access only even or odd addresses)
MOVEQ⁴	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	±16bit s * ±16bit Dn → ±Dn	Multiply signed 16-bit; result: signed 32-bit
MULU	W	s,Dn	-**00	e	-	s	s	s	s	s	s	s	s	s	s	16bit s * 16bit Dn → Dn	Multiply unsig'd 16-bit; result: unsig'd 32-bit
NBCD	B	d	*U*U*	d	-	d	d	d	d	d	d	d	-	-	-	$0 - d_{10} - X \rightarrow d$	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)
OR⁴	BWL	s,Dn	-**00	e	-	s	s	s	s	s	s	s	s	s	s⁴	s OR Dn → Dn	Logical OR
		Dn,d		e	-	d	d	d	d	d	d	d	-	-	-	Dn OR d → d	(ORI is used when source is #n)
ORI⁴	BWL	#n,d	-**00	d	-	d	d	d	d	d	d	d	-	-	s	#n OR d → d	Logical OR #n to destination
ORI⁴	B	#n,CCR	=====	-	-	-	-	-	-	-	-	-	-	-	s	#n OR CCR → CCR	Logical OR #n to CCR
ORI⁴	W	#n,SR	=====	-	-	-	-	-	-	-	-	-	-	-	s	#n OR SR → SR	Logical OR #n to SR (Privileged)
PEA	L	s	-----	-	-	s	-	-	s	s	s	s	s	s	-	↑s → -(SP)	Push effective address of s onto stack
RESET			-----	-	-	-	-	-	-	-	-	-	-	-	-	Assert RESET Line	Issue a hardware RESET (Privileged)
ROL ROR	BWL	Dx,Dy	-**0*	e	-	-	-	-	-	-	-	-	-	-	-		Rotate Dy, Dx bits left/right (without X)
		#n,Dy		d	-	-	-	-	-	-	-	-	-	-	s		Rotate Dy, #n bits left/right (#n: 1 to 8)
	W	d		-	-	d	d	d	d	d	d	d	-	-	-		Rotate d 1-bit left/right (.W only)
ROXL ROXR	BWL	Dx,Dy	***0*	e	-	-	-	-	-	-	-	-	-	-	-		Rotate Dy, Dx bits L/R, X used then updated
		#n,Dy		d	-	-	-	-	-	-	-	-	-	-	s		Rotate Dy, #n bits left/right (#n: 1 to 8)
	W	d		-	-	d	d	d	d	d	d	d	-	-	-		Rotate destination 1-bit left/right (.W only)
RTE			=====	-	-	-	-	-	-	-	-	-	-	-	-	(SP)+ → SR; (SP)+ → PC	Return from exception (Privileged)
RTR			=====	-	-	-	-	-	-	-	-	-	-	-	-	(SP)+ → CCR, (SP)+ → PC	Return from subroutine and restore CCR
RTS			-----	-	-	-	-	-	-	-	-	-	-	-	-	(SP)+ → PC	Return from subroutine
SBCD	B	Dy,Dx	*U*U*	e	-	-	-	-	-	-	-	-	-	-	-	$Dx_{10} - Dy_{10} - X \rightarrow Dx_{10}$	Subtract BCD source and eXtend bit from
		-(Ay),-(Ax)		-	-	-	-	e	-	-	-	-	-	-	-	$-(Ax)_{10} - -(Ay)_{10} - X \rightarrow -(Ax)_{10}$	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	=====	-	-	-	-	-	-	-	-	-	-	-	s	#n → SR; STOP	Move #n to SR, stop processor (Privileged)
SUB⁴	BWL	s,Dn	*****	e	s	s	s	s	s	s	s	s	s	s	s⁴	Dn - s → Dn	Subtract binary (SUBI or SUBQ used when
		Dn,d		e	d⁴	d	d	d	d	d	d	d	-	-	-	d - Dn → d	source is #n. Prevent SUBQ with #n.L)
SUBA⁴	WL	s,An	-----	s	e	s	s	s	s	s	s	s	s	s	s	An - s → An	Subtract address (.W sign-extended to .L)
SUBI⁴	BWL	#n,d	*****	d	-	d	d	d	d	d	d	d	-	-	s	d - #n → d	Subtract immediate from destination
SUBQ⁴	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	*****	e	-	-	-	-	-	-	-	-	-	-	-	Dx - Dy - X → Dx	Subtract source and eXtend bit from
		-(Ay),-(Ax)		-	-	-	-	e	-	-	-	-	-	-	-	-(Ax) - -(Ay) - X → -(Ax)	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	Dn	An	(An)	(An)+	-(An)	(i,An)	(i,An,Rn)	abs.W	abs.L	(i,PC)	(i,PC,Rn)	#n		

Condition Tests (+ OR, ! NOT, ⊕ XOR; ᵘ Unsigned, ᵃ Alternate cc)

cc	Condition	Test	cc	Condition	Test
T	true	1	VC	overflow clear	!V
F	false	0	VS	overflow set	V
HIᵘ	higher than	!(C + Z)	PL	plus	!N
LSᵘ	lower or same	C + Z	MI	minus	N
HSᵘ, CCᵃ	higher or same	!C	GE	greater or equal	!(N ⊕ V)
LOᵘ, CSᵃ	lower than	C	LT	less than	(N ⊕ V)
NE	not equal	!Z	GT	greater than	![(N ⊕ V) + Z]
EQ	equal	Z	LE	less or equal	(N ⊕ V) + Z

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

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, **i** Displacement
BCD Binary Coded Decimal
↑ Effective address
1 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, I, 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