UNIVERSITÉ LAVAL
FACULTÉ DES SCIENCES ET DE GÉNIE
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE

Laboratoire de Radiocommunications
et de Traitement du Signal

ÉTUDES DES DÉFAUTS SYSTÉMATIQUES ET NATURELS DANS LES RÉSEAUX DE COMMUNICATION

PHASE II

ANALYSE DE LA PERFORMANCE DES RÉCEPTEURS QAM EN PRÉSENCE D'UN SIGNAL D'INTERFÉRANCE FM

par

H.T. Huynh, R. Gagnon et D. Angers

pour

Gouvernement du Canada
Ministère des Communications, Ottawa

sous

Contrat OST84-00259
du Ministère des approvisionnements et services

Avril 1985

2 ÉTUDES DES DÉFAUTS SYSTÉMATIQUES ET NATURELS
DANS LES RÉSEAUX DE COMMUNICATIONS

PHASE II

ANALYSE DE LA PERFORMANCE DES RÉCEPTEURS OAM
EN PRÉSENCE D'UN SIGNAL D'INTERFÉRENCE FM

par

Chercheur principal : Dr H.T. Huynh, Ing.
Ingénieur de recherche : R. Gagnon, M.Sc., Ing.
Collaborateur : Dr D. Angers, Ing.

Avril 1985

TABLE DES MATIÈRES

page

page

CHAPITRE I

INTRODUCTION

Le monde de télécommunication se développe à un rythme bousculant. En un siècle, le réseau de communication devient un des produits de l'humanité le plus complexe; il possède des ramifications partout à travers le monde. Jusqu'aux années 60, plus que 90% des systèmes opèrent suivant les vieux principes de communication analogique; particulièrement, les systèmes à micro-onde ils sont souvent de type FDM/FM. Depuis lors, la nouvelle technologie numérique commence à s'imposer très fortement de jour en jour.

La victoire des systèmes numériques est la combinaison de plusieurs facteurs, parmi lesquels les plus importants sont sans doute le faible coût de fabrication et la bonne performance contre les interférences. D'après la projection de BNR, vers les années 90, ces systèmes devraient occuper 90% des réseaux de télécommunications.

Étant donné ce développement fulgurant, la quantité d'information à faire circuler sur les réseaux augmente à une vitesse vertigineuse. Pour pouvoir répondre à ces demandes, il faut construire des systèmes à très grand débit; comme la plupart des bandes de fréquences sont relativement congestionnées, il faut donc introduire des systèmes numériques ayant une efficacité spectrale élevée. Cette nécessité amène des compagnies de l'industrie de communication à proposer des systèmes qui puissent opérer presque dans les mêmes bandes de fréquence que les vieux systèmes analogiques; le dernier né de ce domaine est le système RD-4A de la compagnie "Northern Telecom Limited", disponible sur le marché international depuis 1984. Ce système utilise une modulation en quadrature à 64 niveaux opérant à un rythme de 20 Mbauds à la seconde.

Étant donné la très grande vitesse de ces nouveaux systèmes, leur performance deviendra relativement sensible par rapport à des défauts naturels et systématiques. Pour leur assurer un bon fonctionnement, il faudra alors prendre des précautions sur plusieurs aspects: soigner la conception des composantes électroniques, rendre très directives les antennes émettrices et réceptrices, séparer convenablement les fréquences porteuses. Ce dernier aspect se trouve sous l'autorité du Ministère des communications. À la demande du groupe "Techniques des Systèmes", dirigé alors par M. G. De Couvreur et remplacé par la suite par M. M. Gaudreau, nous avons commencé à examiner l'influence des systèmes FM sur la performance des systèmes de modulation en quadrature (QAM). Cette influence dépend naturellement de l'écart entre les deux porteuses, de leur puissance relative et de leur largeur de bande.

Dans ce rapport, nous présenterons au chapitre II, les points fondamentaux concernant notre problème: modélisation des systèmes QAM, le passage des signaux FM à très grande largeur de bande à travers un récepteur QAM, méthode de calcul de probabilité d'erreur par les moments, analyse de précison; l'ensemble des résultats numériques sera présenté au chapitre III; le rapport sera conclu par le chapitre IV, dans lequel une discussion de l'ensemble des travaux effectués sera présentée.

CHAPITRE II

PASSAGE D'UN SIGNAL FM À TRAVERS UN SYSTÈME QAM

II.1 Introduction

Nous allons présenter dans ce chapitre la méthodologie pour résoudre le problème d'interférence que provoque le passage d'un signal FM à travers un récepteur QAM. Dans cette méthodologie, nous avons besoin d'un modèle précis décrivant le fonctionnement du récepteur dont le comportement face à une interférence FM sera examiné. Ce signal d'interférence étant à très grande largeur de bande, on pourra alors justifier l'hypothèse de quasi-stationnarité pour simplifier l'analyse.

II.2 Modulation en quadrature

La modulation en quadrature est réalisée par la combinaison de deux porteuses en quadrature modulée chacune en amplitude. Soient m et n le nombre de niveaux sur chaque porteuse; on aura donc $M = m \times n$ états possibles pour le signal bidimensionnel résultant. Nous limiterons ce rapport au cas où $m=n=4$, qui est celui de la modulation à 16 états. Sur chaque porteuse, les niveaux sont équidistants, ce qui donne la fig. 2.1.

Les seize points de la constellation correspondent à 16 mots de quatre bits. Ceux-ci sont formés à l'aide de 4 trains binaires groupés deux par deux. Soient A_1, B_1, A_2 et B_2 ces trains. Dans un premier temps, nous avons la formation des mots de deux bits A_1B_1 et A_2B_2 qui serviront à moduler les deux porteuses en faisant correspondre à chaque valeur du mot un niveau de signal avec la correspondance suivante:

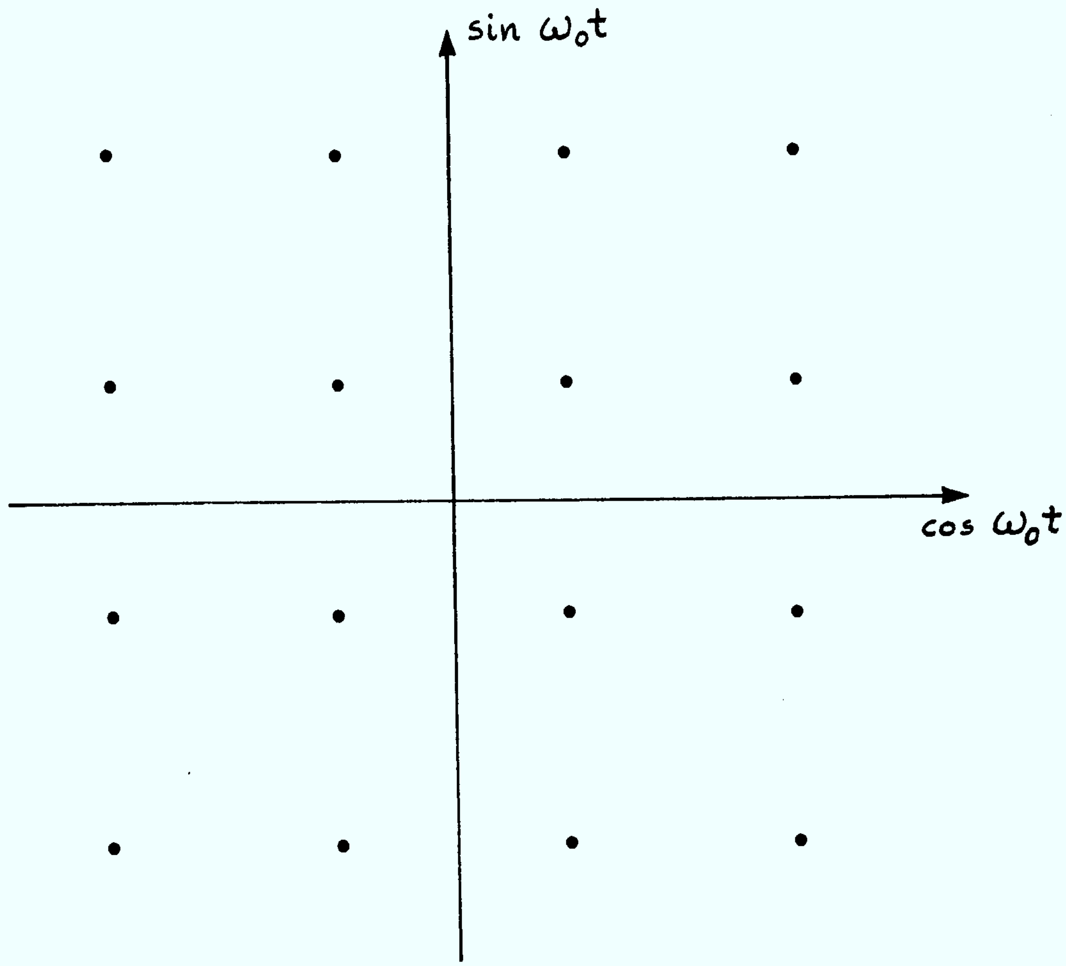


Fig. 2.1 - Constellation QAM-16

A_i B_i	Niveau du signal
1 1	3
0 1	1
0 0	-1
1 0	-3

La correspondance est faite de telle sorte que les mots A_iB_i suivent un code de Gray, i.e. un code tel que deux mots voisins se diffèrent par un seul bit. L'additon des deux porteuses détermine donc 16 états pour les mots A_1B_1, A_2B_2. Pratiquement, ce type de modulation a été retenu pour deux raisons: premièrement, la performance est très acceptable par rapport à l'optimum (perte de 0.2 dB pour une contrainte de puissance moyenne); deuxièment, sa mise en oeuvre est à priori simplifiée du fait qu'elle n'exige pas de technologie particulière.

II.3 Structure du modulateur QAM

La première composante du modulateur est un codeur transformant les quatre trains binaires en deux trains quaternaires et associe à chacun d'eux un signal à quatre niveaux d'amplitude. Ces signaux attaquent le modulateur en fréquence intermédiaire après un filtrage à large bande servant à limiter le spectre. Les deux porteuses modulées sont additionnées pour donner un signal à 16 états. On notera que les deux porteuses en quadrature sont obtenues d'un même oscillateur dans le but de conserver la cohérence entre les phases des deux voies. L'ensemble de ces opérations est montré à la fig. 2.2.

À la sortie du codeur, on trouve sur les voies x et y, des signaux à quatre états d'amplitude déterminés par les combinaisons A_iB_i (i = 1, 2). Cette correspondance est effectuée en suivant un code de Gray montré à la fig. 2.3. Les régions de décisions sont limitées par les droites d'abscisses et d'ordonnées -2,0 et 0,2. Du fait du codage, le franchissement d'un seuil n'occasionne qu'une seule erreur.

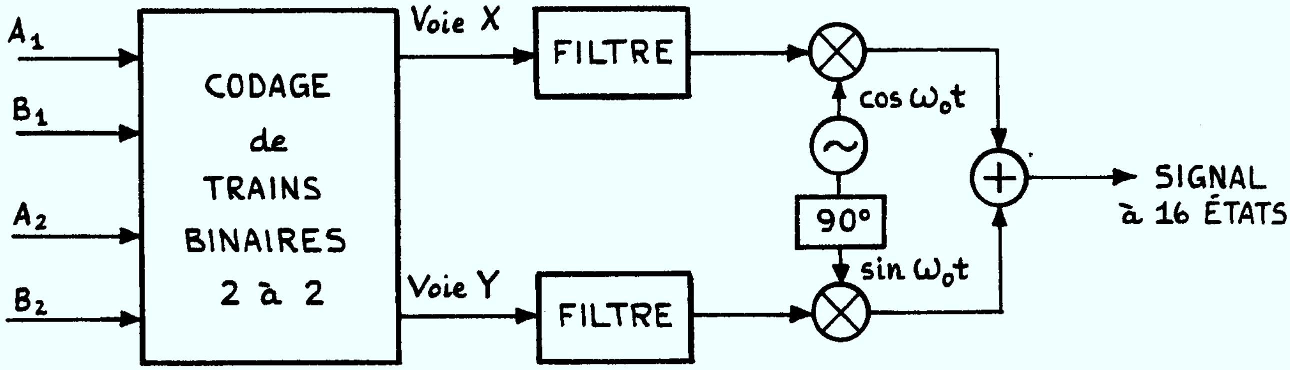


Fig. 2.2 - Structure du modulateur OAM

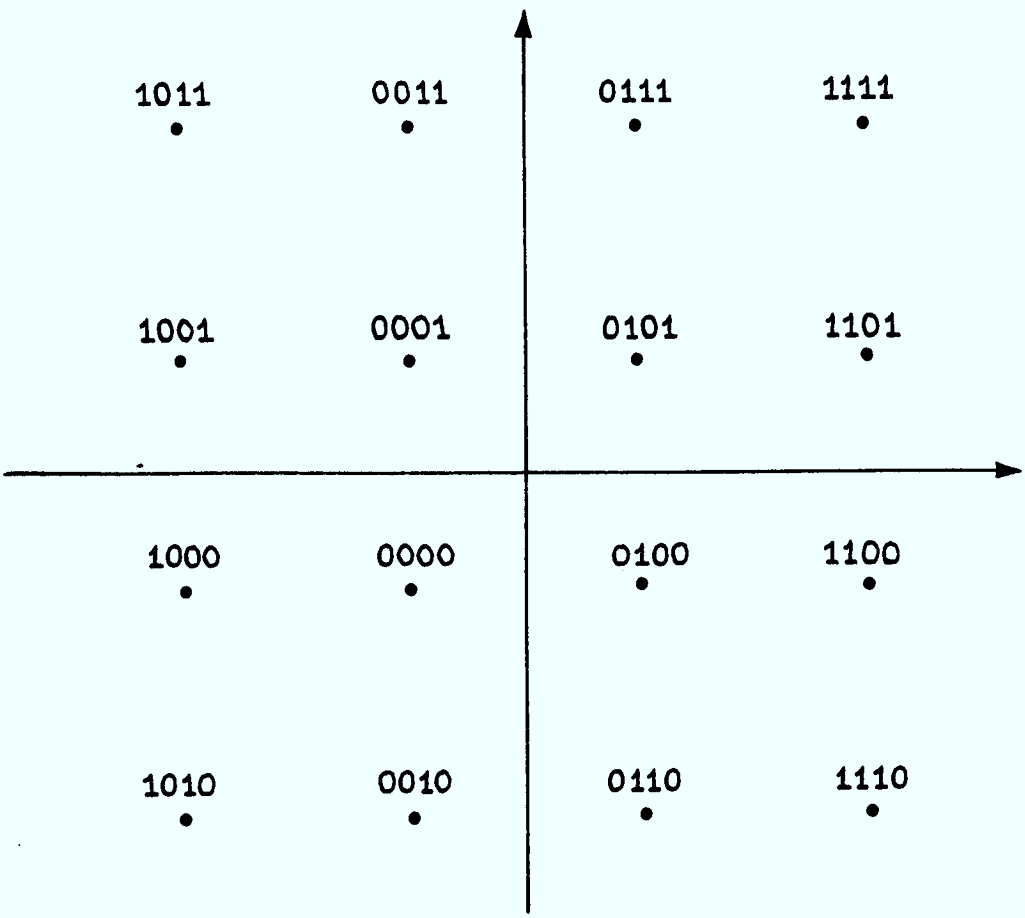


Fig. 2.3 - Code de Gray en QAM

II.4 Formulation du problème

II.4.1 Schéma de la chaîne de transmission

La fig. 2.4 montre l'aspect global des éléments constituant le système de transmission et des différents défauts rencontrés. Le signal à 16 états à la sortie du modulateur attaque un oscillateur micro-onde qui va le transposer en hyperfréquences. La nécessité de placer plusieurs canaux côte à côte conduit à effectuer un filtrage pour limiter leur spectre individuel et pour fixer leur place dans la bande de fréquence de transmission. Cette opération est réalisée par un multiplexage fréquentiel. Après son passage dans le milieu de propagation qui est ici l'espace libre, le signal en hyperfréquence est filtré pour séparer les canaux, puis transposé en fréquence intermédiaire afin d'être démodulé. Le démodulateur suivi du décodeur donne une estimation des trains binaires d'origine et sert à obtenir les références de phase de la porteuse et l'horloge, nécessaire pour une démodulation cohérente et pour le fonctionnement des circuits de décision. Idéalement, l'ensemble de ces opérations nous permettront de reconstituer correctement le signal d'information transmis. Malheureusement, il n'en est rien en pratique, car le signal à fréquence intermédiaire à la réception subit plusieurs effets destructifs suivants:

a) bruit thermique (donc gaussien et blanc),
b) brouillage par les signaux des canaux voisins dans le multiplexeur (donc, le signal FM interférant),
c) "Interférence intersymbole" provoquée par les filtrages,
d) défauts de propagation (fadings lents ou rapides).

L'étude présentée dans ce rapport ne considère que la présence simultanée du bruit thermique, le signal d'interférence FM et les filtres.

II.4.2 Modélisation

Compte tenu des hypothèses précédentes, la chaîne de transmission, définie comme la succession des composantes diverses que rencontre le signal, peut être représentée par la fig. 2.4. Toute la modélisation est effectuée à l'aide de la notion de filtre passe-bas équivalent [10].

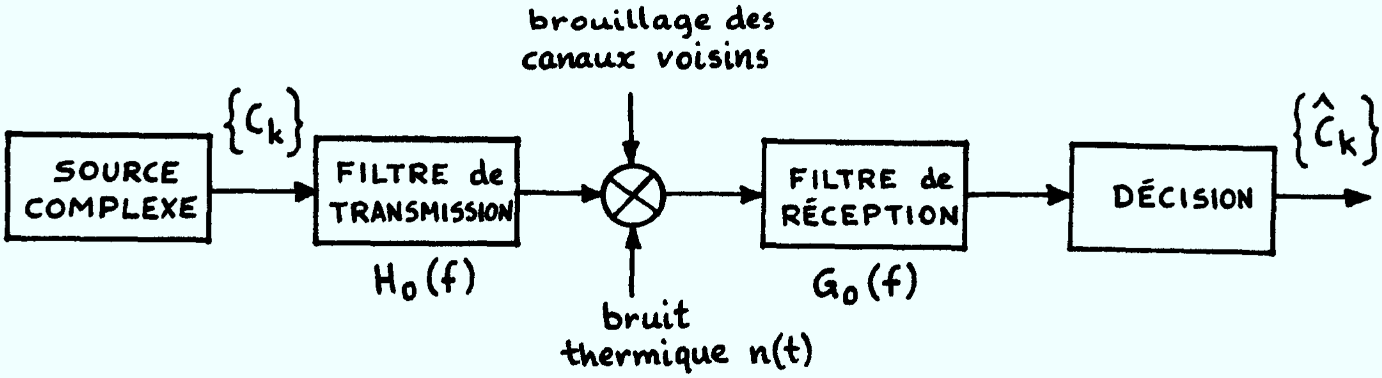


Fig. 2.4 - Système équivalent passe-bas

Source complexe:

Soient $\{a_k\}$ et $\{b_k\}$ les messages quaternaires à l'entrée du modulateur. Il s'agit des trains binaires à la sortie du codeur. Soient $\chi(t)$ et $s(t)$ respectivement l'enveloppe passe-bas et le signal de sortie du modulateur. En général, on a:

$$\chi(t) = \begin{cases} 1 & \text{pour } 0 < t < T \\ 0 & \text{ailleurs} \end{cases} \qquad (1)$$

où T est la durée d'un baud.

$$s(t) = A \sum_{k=-\infty}^{\infty} \{a_k \chi(t-kT)\cos\omega_o t + b_k \chi(t-kT)\sin\omega_o t\} \qquad (2)$$

L'équation (2) peut être mise sous une forme complexe beaucoup plus compacte:

$$s(t) = \mathrm{Re}\{s_e(t)e^{j\omega_c t}\} \qquad (3)$$

où

$$s_e(t) = \sum_{k=-\infty}^{\infty} (a_k + jb_k)\chi(t-kT) \qquad (4)$$

$s_e(t)$ est appelé l'enveloppe complexe de $s(t)$. On est donc amené à définir une source équivalente de messages complexes à 16 états:

$$C_k = a_k + jb_k$$

Les filtres:

Le filtre de transmission $H_o(f)$ rassemble les filtres en hyperfréquences, les filtres vidéo et le milieu de transmission. Le filtre vidéo, de largeur de bande très grande devant les autres, peut être négligé, sa présence n'étant pas nécessaire d'un point de vue théorique. Le filtre de réception, placé à l'entrée du démodulateur a pour fonction de transfert $G_o(f)$. Sa fonction essentielle est de limiter les interférences et le bruit à la réception. Il agit en fréquence intermédiaire, après un préamplificateur de réception.

Le signal voit donc un filtre global de gain complexe $H_o(f)G_o(f)$. Soit $P(t) + jQ(t)$ la réponse de ce filtre au signal $\chi(t)$. En absence de toute interférence et du bruit additif, le signal à l'entrée du démodulateur s'écrit sous la forme complexe suivante:

$$x(t) = A \sum_{k=-\infty}^{\infty} (a_k + jb_k)\left[P(t-kT) + jQ(t-kT)\right] \tag{5}$$

Pour alléger l'écriture, posons:

$$P_k(t) = P(t-kT) \quad \text{et} \quad Q_k(t) = Q(t-kT)$$

il vient:

$$x(t) = \sum_{k=-\infty}^{\infty} (a_k + jb_k)\left[P_k(t) + jQ_k(t)\right]$$

$$x(t) = \sum_{k=-\infty}^{\infty} \left[a_kP_k(t) - b_kQ_k(t)\right] + j\left[a_kQ_k(t) + b_kP_k(t)\right] \tag{6}$$

Si le filtre global H_oG_o possède une parité de l'affaiblissement et du temps de propagation de groupe, i.e., sa fonction de transfert $H_o(f)G_o(f)$ représente une symétrie hermitienne autour de 0, soit:

$$H_o(f)G_o(f) = H_o^*(-f)G_o^*(-f)$$

alors le terme $Q_k(t)$ sera identiquement nul. Dans ce cas, la diaphonie devient nulle et le signal à l'entrée du démodulateur se simplifie comme:

$$x(t) = \sum_{k=-\infty}^{\infty} (a_k + jb_k)P_k(t) \qquad (7)$$

Le bruit thermique:

Le bruit thermique apparaît dans le préamplificateur de réception. Il est additif, centré et à spectre plat dans une bande de fréquences large devant celle du récepteur.

On appellera $N_o/2$ la densité spectrale bilatérale de puissance du bruit. La puissance du bruit σ^2 qui affecte la probabilité d'erreur est donnée par:

$$\sigma^2 = N_oT \int_{-\infty}^{\infty} |G_o(f)|^2 df \qquad (8)$$

où $G_o(f)$ est la fonction de transfert du filtre de fréquence intermédiaire en réception. Il est donc clair qu'un bon choix de $G_o(f)$ peut limiter la puissance du bruit additif dans le récepteur.

Rapport porteuse à bruit

Ce rapport, également appelé le rapport signal-à-bruit, est le paramètre qui affecte le plus la probabilité d'erreur de réception. Il est défini comme le rapport de l'énergie d'un baud sur la densité spectrale de bruit. Étant donné les hypothèses du problème posé par les délégués scientifiques (G. De Couvreur et M. Gaudreau), on adopte tout de suite:

$$H_o(f) = 1$$

Le signal modulé est alors:

$$s(t) = A \sum_{k=-\infty}^{\infty} (a_k \cos\omega_o t + b_k \sin\omega_o t)\chi(t-kT)$$

où A est l'écart entre deux amplitudes voisines de la même voie. L'énergie moyenne par baud (ou par mot transmis) est alors:

$$E_m = \frac{A^2}{16} \left(\sum_{k=1}^{4} \frac{a_k^2 + b_k^2}{2} \right)$$

avec

$$a_k = \pm 1, \pm 3 \quad \text{et} \quad b_k = \pm 1, \pm 3$$

où

$$E_m = 5\, A^2$$

Si l'on s'intéresse à l'énergie crête, on peut adopter la définition:

$$E_c = A^2 \max_{k=1,4} \left\{ \frac{a_k^2 + b_k^2}{2} \right\} = 9\, A^2$$

À partir de ces valeurs, on peut définir les rapports:

$$\frac{E_m}{N_o} \quad \text{et} \quad \frac{E_c}{N_o}$$

qui se ramènent en termes de puissance à l'aide de la durée d'un baud T:

$$\frac{E_m}{N_o} = \frac{E_m/T}{N_o/T} \quad \text{et} \quad \frac{E_c}{N_o} = \frac{E_c/T}{N_o/T}$$

Il s'agit alors des rapports entre la puissance (moyenne ou crête) du signal sur la puissance du bruit dans la bande de Nyquist. Dans la suite du rapport, les résultats seront exprimés en fonction du rapport "Signal à bruit" donné par $\frac{E_m/T}{N_o/T}$ qui repère S/N sur les abscisses des courbes de probabilité d'erreur.

Contrôle automatique de gain

Le contôle automatique de gain (CAG) a pour but de compenser les pertes d'énergie du signal due aux affaiblissements aléatoires qu'il peut subit à travers la chaîne de transmission. Pratiquement parlant, ceci revient à réajuster le gain pour présenter un niveau constant à l'entrée du démodulateur. Dans les calculs analytiques, cette opération permet d'effectuer une normalisation de la réponse du canal par rapport à l'échantillon de niveau maximum. Cette normalisation vise à obtenir des résultats sous une forme universelle.

II.4.3 Le passage d'un signal FM dans un récepteur QAM

Dans le modèle montré à la figure 2.4, le brouillage résulte de la présence d'un (ou de plusieurs) signal FM à très grande largeur de bande. Ce signal brouilleur passe à travers du filtre de réception $G_o(f)$ avant d'être échantillonné au même instant t_o que le signal numérique QAM. Il est clair que son effet à l'entrée du dispositif de décision est de type additif au même titre que le bruit thermique. Pour le cas où ce signal brouilleur possède un indice de modulation élevé, on peut utiliser l'hypothèse "quasi-stationnaire" [6] pour étudier l'effet du filtrage. Il en est connu que dans de telles conditions [5,7], sur un intervalle de temps très court, il se comporte comme un signal sinusoïdal dont la fréquence correspond à la fréquence instantanée du signal FM. Si le signal de bande de base est un processus gaussien, le spectre du signal FM à très grande largeur de bande possède des formes gaussiennes. Ce résultat est connu sous le nom de théorème de Woodward [4]. D'après ces arguments, la distribution $p_B(f)$ de la fréquence instantanée du signal FM admet la forme statistique du signal de modulation. Ainsi:

$$p_B(f) = \frac{1}{\sqrt{2\pi}\ \Delta f_B} \exp\left[-\frac{(f-f_B)^2}{2\Delta f_B^2}\right] \tag{9}$$

où Δf_B est la déviation efficace (i.e. la valeur rms) de la fréquence instantanée par rapport à la porteuse FM. La sortie du filtre, sous hypothèse quasi-stationnaire est donc un signal sinusoïdal dont l'amplitude est amplifiée par le gain $|G_o(f)|$ et sa phase est quelconque sans référence à l'égard de l'instant d'échantillonnage. De cette manière, la composante due au brouilleur FM, à l'entrée du dispositif de décision est finalement de la forme:

$$I = B \cos \theta \tag{10}$$

où B est l'amplitude du signal FM, amplifiée par le gain G(f) et θ une variable aléatoire uniformément répartie entre 0 et 2π. Comme on vient de le constater, l'hypothèse "quasi-stationnaire" permet de simplifier sérieusement le problème.

En plus de tenir compte de cette simplification, nous considérons également une autre hypothèse un peu plus complexe: pour une interférence à très grande largeur de bande, son influence est équivalente à une autre interférence de même densité spectrale de puissance. À prime abord, cette hypothèse semble présenter une faiblesse remarquée: l'effet de corrélation entre les phases des composantes spectrales est négligé. Toutefois, on se rend compte, physiquement parlant, que ces phases ne pourraient jamais être très cohérentes car leur puissance totale est imposée par la densité spectrale globale. Tenant compte de cette contrainte sur la puissance totale, cette équivalence représente donc une avenue remarquable. En effet, on pourra construire un ensemble de signaux harmoniques situés à des fréquences quelconques tels que montrés à la fig. 2.5.

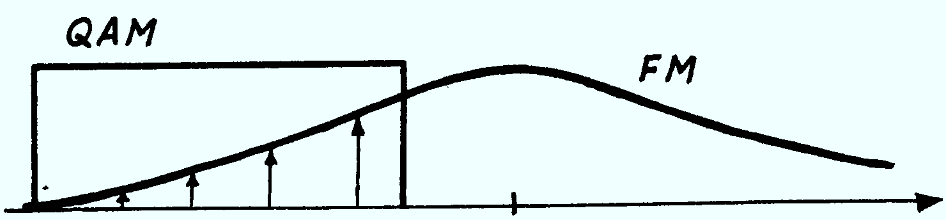


Fig. 2.5 - Considérations spectrales du signal FM
par rapport au système QAM

L'ensemble de ces signaux sinusoïdaux possède des amplitudes déterminées par la densité spectrale du signal FM, sous contrainte que leur puissance totale soit égale à la puissance partielle du signal d'interférence, passant à travers le filtre $G_0(f)$. Les résultats obtenus par ces deux approches "quasi-stationnarité" et "équivalence de puissance" représenteraient les deux limites possibles pour la performance globale de la probabilité d'erreur. À cause du théorème de la limite centrale, l'hypothèse de l'équivalence de puissance sera parfois appelée l'hypothèse gaussienne.

II.5 Calcul de la probabilité d'erreur

II.5.1 Forme générale

Nous allons nous placer dans la dernière partie de la chaîne de transmission, à la sortie du filtre $G_o(f)$; le signal observé s'écrit:

$$x(t) = \sum_{k=-\infty}^{\infty} \{a_k P_k(t) - b_k Q_k(t)\} + j\{b_k P_k(t) + a_k Q_k(t)\}$$

$$+ B_x(t) + jB_y(t) + U(t) + jV(t) \tag{11}$$

Le démodulateur suivi de l'échantillonneur examine, aux instants $t_k = t + kT$, les parties réelle et imaginaire de $x(t)$. À l'instant τ, on a:

$$\mathrm{Re}\{x(t)\} = \sum a_k P_k(\tau) - b_k O_k(\tau) + I_x(\tau) + U(\tau)$$

$$\mathrm{Im}\{x(\tau)\} = \sum\{b_k P_k(\tau) + a_k Q_k(\tau)\} + I_y(\tau) + V(\tau)$$

Nous pouvons, d'autre part, séparer $x(\tau)$ en cinq termes:

$$x(\tau) = (a_o + jb_o)P_o(\tau) - (b_o - ja_o)Q_o(\tau)$$

$$+ \sum_{k\neq 0} \{a_k P_k(\tau) - b_k Q_k(\tau)\} + j\{a_k Q_k(\tau) + b_k P_k(\tau)\}$$

$$+ B_x(\tau) + jB_y(\tau) + U(\tau) + jV(\tau) \tag{12}$$

Ces différents termes représentent respectivement:

- le signal utile
- le signal de diaphonie qui traduit le couplage entre les deux voies x et y

- l'interférence intersymbole $II(\tau)$
- le brouilleur $B(\tau)$
- le bruit additif.

La somme de $II(\tau)$ et $B(\tau)$ peut être mise sous la forme:

$$I(\tau) = II(\tau) + B(\tau) = X(\tau) + jY(\tau)$$

X et Y sont des variables aléatoires centrées de même loi de distribution F(I) symétrique autour de 0. $P_k(\tau)$ et $Q_k(\tau)$ sont des termes déterministes, a_k et b_k des variables aléatoires indépendantes et prenant quatre états équiprobables.

À chaque point (i), représentatif d'un des 16 états, est associée une région de décision D_i, au sens de critère du maximum de vraisemblance. La probabilité d'erreur sur les états s'écrit:

$$P_e = \sum_{i=1}^{16} P_i P\{r \varepsilon D_i | i \text{ soit émis}\}$$

où P_i est la probabilité à priori de l'état (i) et r l'observation (le signal à la sortie du récepteur). Les 16 états étant supposés équiprobables, on a $P_i = 1/16$ et Pe se simplifie:

$$P_e = \frac{1}{16} \sum P\{r \varepsilon D_i | i \text{ soit émis}\}$$

En présence de l'interférence I, P_e devient:

$$P_e = \int_{II} \sum P_i P\{r \varepsilon D_i \big| i \text{ soit émis et } I = b\} dF(b)$$

où II est l'ensemble de toutes les valeurs possibles que peut prendre $I(\tau)$. Le problème consiste donc à évaluer la probabilité d'erreur conditionnelle à l'émission de (i) et pour une valeur donnée de $I = X + jY$. Nous allons en fait évaluer directement les probabilités d'erreur sur chaque train binaire et non pas sur le signal global. Ce sont, en effet, des mesures sur les trains estimés qui sont réalisées dans la pratique. Nous verrons néanmoins comment relier les probabilités sur les trains élémentaires à la probabilité globale. Normalisons d'abord le signal reçu par rapport à $P_O(\tau)$:

$$x_N(\tau) = (a_o + jb_o) - (b_o - ja_o)\frac{Q_o(\tau)}{P_o(\tau)} + \frac{X+jY}{P_o(\tau)} + \frac{U+jY}{P_o(\tau)}$$

Pour ne pas alourdir les notations, nous réutiliserons $x(\tau)$ à la place de $x_N(\tau)$, ce qui revient à imposer $P_O(\tau) = 1$. Soit:

$$x(\tau) = (a_o + jb_o) - (b_o - ja_o)Q_o(\tau) + (X + jY) + (U + jV)$$

Posons $\alpha = - b_oQ_o(\tau) + X$ et $\beta = a_oQ_o(\tau) + Y$.

Nous appellerons $P_e^{A_1}$ (respectivement $P_e^{B_1}$) la probabilité d'erreur sur le train A_1 (respectivement B_1). On peut démontrer (voir Appendice 1) que:

$$P_e^{A_1} = \frac{1}{4} \sum_{b_o=\pm 1, \pm 3} \frac{1}{2} \int_{-\infty}^{\infty} \operatorname{erfc}\left(\frac{A+\alpha}{\sigma\sqrt{2}}\right) dF(x)$$

$$P_e^{B_1} = \frac{1}{2} P_e^{A_1}$$

Comme nous avons discuté précédemment, l'interférence intersymbole et la diaphonie ne sont pas les éléments importants dans notre travail, on peut donc choisir le cas optimum pour notre analyse, i.e., le cas des filtres à symétrie hermitienne. Il vient:

$$Q(t) \equiv 0$$

et alors:

$$P_e^{A_1} = \frac{1}{2} \int_{-\infty}^{\infty} \operatorname{erfc}\left(\frac{A+X}{\sigma\sqrt{2}}\right) dF(x) \qquad (13)$$

$$P_e^{B_1} = \frac{1}{2} P_e^{A_1}$$

En absence de toute forme d'interférence, les résultats se simplifient pour donner:

$$P_e^{A_1} = \frac{1}{2}\,\mathrm{erfc}\left(\frac{A}{\sigma\sqrt{2}}\right)$$

$$P_e^{B_1} = \frac{1}{2}\,P_e^{A_1}$$

Cela revient à supposer que le canal est idéal (canal de Nyquist). Ces dernières expressions de la probabilité d'erreur servent alors à calculer les performances en présence du bruit gaussien additif seul. Toutefois, pour ce canal idéal qui élimine complètement l'interférence intersymbole, le problème de calcul du brouilleur deviendrait nettement plus complexe; c'est pourquoi nous tiendrons compte dans la suite de l'interférence intersymbole que provoque un filtre de Butterworth à phase linéairement égalisée.

II.5.2 Calcul d'intégrale par la méthode de Ho et Yeh

Dans l'expression (12), nous avons à calculer l'intégrale:

$$J = \int_{-\infty}^{\infty} \mathrm{erfc}\left(\frac{A+X}{\sigma\sqrt{2}}\right)dF(X) \qquad (14)$$

La méthode de Ho et Yeh consiste à développer la fonction erfc en série de Taylor au voisinage du point $A/\sigma\sqrt{2}$:

$$\mathrm{erfc}\left(\frac{A+x}{\sigma\sqrt{2}}\right) = \mathrm{erfc}\left(\frac{A}{\sigma\sqrt{2}}\right) + \sum_{k=1}^{\infty}\left(\frac{x}{\sigma\sqrt{2}}\right)^k \frac{1}{k!}\,D_k$$

avec

$$D_k = \left.\frac{d^k \mathrm{erfc}(x)}{dx^k}\right|_{x = \frac{A}{\sigma\sqrt{2}}} \qquad (15)$$

D'où:

$$J = \int_{-\infty}^{\infty} \mathrm{erfc}\left(\frac{A}{\sigma\sqrt{2}}\right)dF(x) + \sum_{k=1}^{\infty}\frac{1}{k!}\,D_k \int_{-\infty}^{\infty}\left(\frac{x}{\sigma\sqrt{2}}\right)^k dF(x)$$

Cette expression se simplifie en remarquant que $\int_{-\infty}^{\infty} dF(x) = 1$ et en posant $M_k = \int_{-\infty}^{\infty} x^k dF(x)$:

$$J = \operatorname{erfc}\left(\frac{A}{\sigma\sqrt{2}}\right) + \sum_{k=1}^{\infty} \frac{1}{k!} D_k M_k \frac{1}{(\sigma\sqrt{2})^k}$$

Notons tout de suite que D_k est lié aux polynômes de Hermite $H_n(x)$ [8]:

$$\frac{d^k}{dx^k} \operatorname{erfc}(x) = (-1)^k \frac{2}{\sqrt{\pi}} H_{k-1}(x)\exp(-x^2)$$

Finalement:

$$J = \operatorname{erfc}\left(\frac{A}{\sigma\sqrt{2}}\right) + \frac{2}{\pi} \exp\left(-\frac{A^2}{2\sigma^2}\right) \sum_{k=1}^{\infty} \frac{1}{k!(\sigma\sqrt{2})^k} H_{k-1}\left(\frac{A}{\sigma\sqrt{2}}\right) M_k \qquad (16)$$

Les paramètres M_k dans (16) s'appellent les moments d'ordre k de la variable aléatoire X.

Calcul des moments M_k

Soient X et F(x) respectivement une variable aléatoire et sa fonciton de distribution de probabilité. Par définition, son moment d'ordre k est donné par:

$$M_k = \int_{-\infty}^{\infty} x^k dF(x)$$

Ces moments peuvent être obtenus à partir de sa fonction caractéristique $\phi_X(\omega)$ qui est la transformée de Fourier de sa densité de probabilité:

$$\phi_X(\omega) = E\{e^{j\omega X}\} = \int_{-\infty}^{\infty} e^{j\omega x} dF(x)$$

Dérivant successivement $\phi_X(\omega)$, on obtient:

$$\phi_X(\omega) = 1 + j\omega M_1 + \ldots + (j\omega)^k \frac{M_k}{k!} + \ldots$$

Ainsi, si l'on arrive à calculer la fonction caractéristique $\phi_X(\omega)$ de la variable aléatoire X, la somme de toutes les interférences présentes sur la voie X, i.e.:

$$X = \sum_{k \neq 0} a_k P(\tau - kT) + B_X(\tau) \tag{17}$$

Comme $B_X(\tau)$ et toutes les variables a_k sont statistiquement indépendantes, la fonction caractéristique est tout simplement le produit de toutes les fonctions caractéristiques correspondant à chacun des termes de l'équation (17). Dans leur travail, Ho et Yeh ont proposé une méthode récursive permettant de calculer les moments d'ordre successif. Malheureusement, cette récursivité est très sensible par rapport à l'accumulation des erreurs d'arrondissement; c'est ainsi qu'elle aboutit très souvent à des moments négatifs d'ordre pair, ce qui entre en contradiction avec la définition mathématique [9]. Ce phénomène nous exige à calculer les moments par la méthode directe; comme il s'agit d'une méthode "exhaustive", elle consommera évidemment plus de temps d'ordinateur. Avant de présenter cette méthode de calcul, observons que les variables aléatoires présentes dans (17) sont toutes statistiquement indépendantes; il est alors connu que les moments de leur somme sont des combinaisons de leurs moments. En effet, il suffit de considérer X comme la somme de deux variables aléatoires X_1 et X_2 statistiquement indépendantes et ayant les densités de probabilité paires; il vient immédiatement:

$$X = X_1 + X_2$$

$$E\{X^k\} = E\{(X_1 + X_2)^k\}$$

$$= \sum_{\ell=0}^{k} C_\ell^k \, E\{X_1^\ell\} E\{X_2^{k-\ell}\}$$

X_1 et X_2 étant symétriquement distribuées, on a:

$$E\{X_1^{2\ell+1}\} = E\{X_2^{2\ell+1}\} = 0$$

ce qui implique immédiatement que les moments d'ordre impair $E\{X^{2\ell+1}\}$ sont identiquement nuls. Seuls les moments d'ordre pair $M_{2k} = E\{X^{2k}\}$ existent et sont donnés par:

$$M_{2k} = \sum_{p=0}^{2k} C_{2p}^{2k} M_{2p}^{1} M_{2k-2p}^{2} \tag{18}$$

où M_{2p}^{1} et M_{2k-2p}^{2} sont respectivement les moments d'ordre 2p et les moments d'ordre 2k-2p des variables aléatoires X_1 et X_2. L'équation (18) est donc la clef de la méthode directe de calcul des moments d'une somme des variables aléatoires indépendantes.

Ainsi, pour appliquer la méthode directe à l'étude de l'influence d'un signal FM à très grande largeur de bande, on n'a qu'à calculer séparément les moments de l'interférence intersymbole et ceux du signal FM. Ces derniers se calculent de manière très simple grâce au modèle proposé par l'équation (10). Quant aux moments de l'interférence intersymbole, ils font l'objet du paragraphe suivant.

II.5.3 Méthode directe de calcul des moments [2]:

Soit U le nombre d'échantillons de la réponse du canal à un créneau $\chi(t)$, l'enveloppe passe-bas d'un baud, telle que décrite par (1). Il y a donc U-1 termes d'interférence intersymbole; le signal émis pouvant prendre l'une des quatre valeurs (±1, ±3), le nombre N de configurations possibles est:

$$N = 4^{U-1}$$

La variable aléatoire "Interférence intersymbole" X s'écrit:

$$X = \sum_{k\neq 0} a_k P_k$$

où

$$P_k = P(\tau+kT)$$

Les moments M_{2p} de X deviennent:

$$M_{2p} = \sum_{a_k} P[X_k = X]X^{2p}$$

$$M_{2p} = \frac{1}{N} \sum_{a_k} \left(\sum_{k\neq 0} a_k P_k \right)^{2p}$$

Le calcul des M_{2p} étant ici indépendant pour chaque élément, la précision peut être choisie arbitrairement.

Il nous reste maintenant à prouver la convergence de la méthode et à déduire les critères permettant de vérifier celle-ci. Du fait de l'utilisation des ordinateurs, nous sommes amenés à effectuer des troncations à divers niveaux pour le calcul des séries numériques. Tenant compte de la parité de dF(x), l'intégrale J s'écrit:

$$J = \operatorname{erfc}\left(\frac{A}{\sigma\sqrt{2}}\right) + \frac{2}{\sqrt{\pi}} \exp\left(-\frac{A^2}{2\sigma^2}\right) \sum_{k=1}^{\infty} \frac{M_{2k}}{(2k)!} \times \frac{H_{2k-1}\left(\frac{A}{\sigma\sqrt{2}}\right)}{(\sigma\sqrt{e})^{2k}}$$

Soit U_k le terme général de cette série. Il vient:

$$\frac{U_{k+1}}{U_k} = \frac{M_{2k+2}}{M_{2k}} \times \frac{H_{2k+1}\left(\frac{A}{\sigma\sqrt{2}}\right)}{H_{2k-1}\left(\frac{A}{\sigma\sqrt{2}}\right)} \times \frac{1}{\left(\sigma\sqrt{2}\right)^2} \times \frac{1}{(2k+1)(2k+2)} \qquad (19)$$

Pour k très grand, on a [8]:

$$\left|H_{2k-1}(x)\right| < \frac{2^k}{\sqrt{2}} (2k-3)!! \sqrt{2k-1} \exp\left(\frac{x^2}{2}\right)$$

où

$$(2k-3)!! = (2k-3)(2k-1)\ldots 3.1$$

Alors, l'équation (19) se simplifie pour donner:

$$\left|\frac{U_{k+1}}{U_k}\right| \simeq \frac{M_{2k+2}}{M_{2k}} \frac{(2k-1)!!}{(2k-3)!!} \frac{1}{(\sigma\sqrt{2})^2} \frac{1}{(2k+1)(2k+2)} \sqrt{\frac{2K+1}{2K-1}} \qquad (20)$$

D'autre part, on a

$$M_{2k} = \int_{-\infty}^{\infty} X^{2k} dF(x) < \int_{-\infty}^{\infty} [\mathrm{Sup}(x)]^{2k} dF(x)$$

ou

$$X = \sum_{\ell \neq 0} a_\ell P(\tau - kT)$$

d'où

$$\mathrm{Sup}(x) = 3 \sum_{\ell \neq 0} \left|P(\tau-\ell T)\right|$$

alors

$$\frac{M_{2k+2}}{M_{2k}} < \left(3 \sum_{\ell \neq 0} \left|P(\tau-\ell T\right|\right)^2$$

L'équation (20) devient:

$$\left|\frac{U_{k+1}}{U_k}\right| < \left(\frac{3 \sum_{\ell \neq 0} \left|P(\tau-\ell T)\right|}{\sigma\sqrt{2}}\right)^2 \sqrt{\frac{2k-1}{2k+1}} \frac{1}{k+1}$$

Le critère d'Alembert nous assure la convergence pour:

$$3 \sum_{\ell \neq 0} \left|P(\tau-\ell T)\right| < \sigma\sqrt{2}$$

Cette condition n'est réalisée que si le canal présente des distorsions relativement faibles. De manière pratique, cette convergence est liée à la précision de la méthode, qui est représentée par l'erreur R_N due à la troncation de la série à partir du $N^{\text{ième}}$ terme:

$$R_N = \sum_{k=N}^{\infty} \frac{M_{2k}}{(2k)!} \frac{H_{2k-1}(\frac{A}{\sigma\sqrt{2}})}{(\sigma\sqrt{2})^{2k}} \frac{2}{\sqrt{\pi}} \exp(-\frac{A^2}{2\sigma^2})$$

Pour étudier le comportement de R_N, nous allons utiliser des majorants de M_{2k} et $H_{2k-1}(x)$ comme précédemment, et remplacer la factorielle pour l'approximation de Sterling:

$$k! \sim k^k e^{-k} \sqrt{2\pi K}$$

Il vient:

$$R_N < \sqrt{\frac{2}{\pi}} \exp(\frac{-A^2}{4\sigma^2}) \sum_{k=N}^{\infty} (\frac{3 \sum_{\ell} |P(\tau-kT)|}{\sigma\sqrt{2}\ \sqrt{k}})^{2k} \times \frac{1}{\sqrt{2K-1}} \frac{1}{k^k e^{-k} \sqrt{2\pi k}} \quad (21)$$

Comme

$$\sqrt{2\pi N} < \sqrt{2\pi k}$$

(21) peut se réécrire:

$$R_N < \sqrt{\frac{2}{\pi}} \exp(\frac{-A^2}{4\sigma^2}) \sum_{k=N}^{\infty} (\frac{3 \sum_{\ell \neq 0} |P(\tau-\ell T)|}{\sigma\sqrt{2k}})^{2k} \times \frac{1}{e^{-k}} \frac{1}{\sqrt{2N-1}\ \sqrt{2\pi N}} \quad (22)$$

En posant $k = p + N$, la série de (22) devient:

$$\frac{1}{e^{-N}} (\frac{3 \sum_{\ell \neq 0} |P(\tau-\ell T)|}{\sqrt{N}\ \sigma\sqrt{2}})^{2N} \sum_{p=0}^{\infty} [(\frac{3 \sum_{\ell \neq 0} |P(\tau-\ell T)|}{\sqrt{N}\ \sigma\sqrt{2}})^2 \frac{1}{e^{-1}}]^p \quad (22')$$

La forme de (22') est une série géométrique dont le résultat est bien connu; ainsi, la borne supérieure de R_N définie par (22) s'obtient comme:

$$R_N < \sqrt{\frac{2}{\pi}} \frac{\exp(\frac{-A^2}{4\sigma^2})}{N!\ \sqrt{2N-1}} \frac{\overline{M_N}}{(\sigma\sqrt{2})^{2N}}$$

où

$$\overline{M_N} = \frac{\left(3 \sum_{\ell \neq 0} \left|P(\tau - \ell T)\right|\right)^{2N}}{1 - e\left(\frac{3 \sum_{\ell \neq 0} \left|P(\tau - \ell T)\right|}{\sqrt{N}\ \sigma\sqrt{2}}\right)^2} \tag{23}$$

Cette majoration de l'erreur de troncation, réalisée sur ordinateur, en même temps que le calcul des moments tronqués, servira donc de manière pratique comme critère d'arrêt dans le calcul de l'interférence intersymbole. Ainsi, la série est arrêtée quand l'erreur relative atteint 10^{-5} dans notre programme.

II.6 Conclusion

Nous avons présenté dans ce chapitre, et ce, de manière très complète, la méthodologie permettant d'analyser la performance d'un récepteur QAM en présence d'un signal d'interférence FM. Il consiste à considérer cette interférence comme tout simplement une composante additive à l'entrée du dispositif de décision. Ainsi, la méthode qu'avaient proposée Ho et Yeh pour étudier les systèmes BPSK peut être adaptée à la résolution de notre problème. Toutefois, la récursivité de leur méthode est très sensible vis-à-vis l'erreur de troncation; ce qui nous amène à calculer directement les moments de l'interférence globale. Cette interférence, qui est une somme de plusieurs variables aléatoires indépendantes, peut être analysée systématiquement. Il est intéressant de noter que la précision de cette méthode directe est contrôlable par un critère quantitatif préalablement défini.

Malgré un temps de calcul assez important, la flexibilité de la méthode nous permet de tenir compte des interférences de toute nature, ce qui représente un avantage remarquable par rapport à d'autres méthodes connues dans la littérature. L'exploitation de cette méthode, ainsi que sa programmation seront présentées dans le chapitre suivant.

CHAPITRE III

PRÉSENTATION DES RÉSULTATS ET DISCUSSION

III.1 Méthode des moments

III.1.1 Introduction

Les débuts de nos travaux ont porté d'abord sur la mise au point d'un programme en PASCAL réalisant le calcul de la probabilité d'erreur dans un récepteur QAM par la méthode de Ho et Yeh [1].

Cette méthode, que l'on appellera méthode des moments, a été développée par les auteurs en question pour le calcul de la probabilité d'erreur en présence d'IIS. Toutefois, il a été souligné [2] que cette méthode est suffisamment flexible pour permettre le calcul de P_e en présence d'une ou même de plusieurs interférences quelconques. Dans ce dernier cas, dans l'hypothèse que les différentes interférences sont indépendantes, il suffit de calculer les moments de chaque interférence séparément puis de les combiner par une formule appropriée.

Pour vérifier le bon fonctionnement des programmes développés, nous avons calculé P_e en présence d'IIS puis d'une interférence sinusoïdale et avons comparé nos résultats avec ceux publiés [3]. Ensuite, une fois cette étape réalisée, nous avons entrepris le calcul de P_e en présence d'une interférence FM comme nous nous proposions de le faire dans la phase préliminaire.

III.1.2 Calculs des moments de l'IIS

III.1.2.1 Méthode de calcul

L'interférence intersymbole (IIS) apparaît à la réception lors de la transmission de plusieurs symboles consécutifs et résulte du fait que la réponse du canal n'est pas complètement nulle à des multiples de la période d'échantillonnage. C'est une variable aléatoire de la forme:

$$X(t_o) = \sum_{i \neq 0}^{\infty} a_i P(t_o - iT) \qquad \text{(voie en phase)}$$

où p(t) est la partie réelle de la réponse passe-bas du canal à un créneau (ou à toute autre forme de "pulse"), et a_i est le symbole transmis à l'instant $t_o - iT$ dont les valeurs possibles dépendent du nombre de nouveaux du QAM et sont équiprobables. On désire calculer les moments d'ordre 2k de X. Donc:

$$E\{X^{2k}\} = E[\sum_{i \neq 0}^{\infty} a_i P(t_o - iT)]^{2k}$$

Nous avons réalisé ce calcul par la méthode directe qui consiste à moyenner sur toutes les valeurs possibles de X, bien qu'il existe d'autres méthodes disponibles. En pratique, la sommation infinie est tronquée à U termes de manière à minimiser les calculs. Toutefois, il faut être certain que la convergence des moments est atteinte. Ce calcul se réalise sans trop de mal si les différents échantillons de la réponse du canal décroissent rapidement. Si ce n'est pas le cas, la quantité énorme de calcul peut rendre la tâche impossible. Si on prend un filtre Butterworth d'ordre 5 et une "pulse" rectangulaire, 17 échantillons suffisent pour permettre la convergence des moments. Malgré cela, la quantité de calcul demeure considérable. Par exemple, si on prend le QAM-64 on aura $8^{16} = 2.8 \times 10^{14}$ valeurs de X à moyenner. Ce nombre est encore trop élevé pour que les résultats sortent en un temps raisonnable. Heureusement, il est possible de diminuer les temps de calcul si on tient compte du fait que les différents symboles transmis sont indépendants. En effet, si on sépare la variable X en deux variables indépendantes

X_1 et X_2 ayant chacune U-1/2 termes, elles-mêmes pouvant être séparées en 2 autres variables , on aura pour le calcul des moments:

$$E\{X^{2k}\} = E\{(X_1 + X_2)^{2k}\} = \sum_{j=0}^{k} C_{2k}^{2j} \; E\{X_1^{2j} X_2^{2k-2j}\}$$

et

$$E\{X_1{}^{2j} X_2^{2k-2j}\} = E\{X_1^{2j}\} \; E\{X_2^{2k-2j}\}$$

car X_1 et X_2 sont indépendantes. On pourra donc utiliser cette formule pour combiner les moments de X_1 et X_2 ou pour combiner les moments de n'importe quelles interférences à la condition qu'elles soient indépendantes. Par exemple, on pourra, à l'aide de cette formule, combiner sans difficulté l'IIS avec une interférence sinusoïdale comme on le verra dans la section suivante.

III.1.2.2 Description de QAM

Le calcul de l'IIS par la méthode directe et le calcul de P_e par la méthode des moments ont été réalisés par le programme QAM. L'organigramme simplifié de ce programme est présenté à la figure 3.1. Les résultats de ces premiers essais ont été comparés avec ceux présentés dans [3] pour le cas des QAM-4, 16, 36 et 64 avec un filtre Butterworth ordre 5 et une "pulse rectangulaire". La convergence des moments a été véfifiée pour le cas précis discuté. Toutefois, si on voulait changer l'allure du filtre ou de la "pulse", il faudrait vérifier à chaque fois que la convergence est atteinte. Si elle n'est pas atteinte, il faudra augmenter le nombre d'échantillons de la réponse du canal et par conséquent, les temps de calcul augmenteront proportionnellement. Toutefois, il faudra à chaque fois modifier les procédures de calcul des moments IIS. Le programme QAM imprime deux valeurs de P_e. P_{E1} est la probabilité d'erreur en présence de bruit gaussien seulement et P_{E2} est la probabilité d'erreur avec bruit gaussien et IIS. Ces résultats dépendent des trois paramètres: BT, N et SNR. BT est la largeur de bande du filtre normalisé à 1/T. N est le paramètre qui sélectionne le type de QAM utilisé où $M = (2N)^2$ et M est le nombre d'états. Finalement, SNR est le rapport de la puissance moyenne du signal QAM à la puissance du bruit en dB.

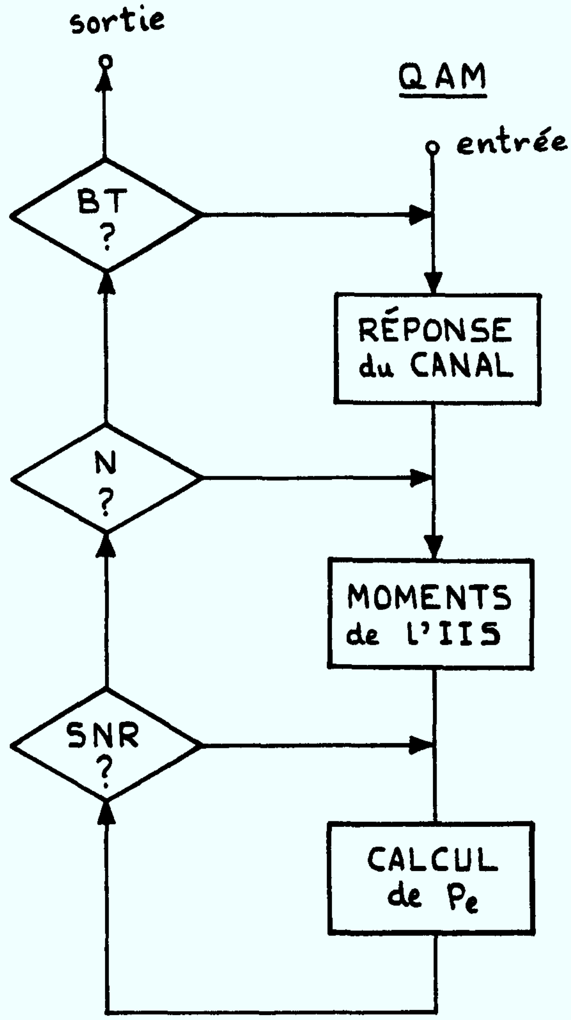


Fig. 3.1 - Organigramme de QAM

$$SNR = 10 \log_{10} \frac{P_m}{N_o/2T} \qquad \text{où} \qquad P_m = \frac{A^2}{T} \frac{\left((2N)^2-1\right)}{3}$$

Le calcul des moments IIS pour les types de modulation QAM-4, 16, 36 et 64 est réalisé par les procédures MOM4, MOM16, MOM36 et MOM64 en utilisant la méthode directe telle que mentionnée. On réalise numériquement le calcul de la réponse du canal par la méthode d'intégration de Romberg par les fonctions REP, ROM et FREQ. Ces fonctions réalisent l'intégrale suivante:

$$s(kT) = 2 \int_0^L \left(\frac{\sin \pi\Omega}{\pi\Omega}\right)\left(\frac{1}{\left(1+(2\Omega/BT)^{10}\right)^{\frac{1}{2}}}\right) \cos 2\pi k\Omega \, d\Omega$$

où L est choisi afin de limiter les erreurs et les temps de calcul. La précision du résultat est fixée dans ROM → à 10^{-12} ou par le nombre d'ittérations (15). On peut modifier cette précision si on veut gagner sur les temps de calcul. De plus, on peut modifier l'allure de la réponse du canal en modifiant la fonction à intégrer qui est dans FREQ. Dans cette fonction, la variable d'intégration est normalisée à 1/T et $B = 2f_c$. Quant au calcul de P_e avec IIS (calcul de la série), il est réalisé dans HERMITE. Cette fonction nécessite que les moments IIS jusqu'à un certain ordre 2K soient calculés au préalable. Donc, il faut choisir le paramètre FIN, déterminant le nombre de moments à calculer, suffisamment grand pour permettre la convergence de la série dont la précision a été fixée à 10^{-5}. La fonction HERMITE réalise le calcul suivant:

$$y = \frac{e^{-x^2}}{\sqrt{\pi}} \sum_{k=1}^{N} E_k M_{2k}$$

où

$$E_k = \frac{x^{2k}}{(2k)!} \left[\left(1 - \frac{4k-5}{2x^2}\right)E_{k-1} - \frac{(k-2)}{(k-1)} E_{k-2}\right]$$

avec

$$E_1 = x^3 \qquad \text{et} \qquad E_2 = \frac{x^4}{24} (8x^3 - 12x)$$

L'argument x est donné par:

$$x = \frac{Ap_o}{\sqrt{2}\ \sigma}$$

On calcule A à partir de SNR et σ^2 est la puissance du bruit à l'entrée du dispositif de décision. P_o est la valeur de la réponse du canal à l'instant d'échantillonnage. Donc, en combinant les équations suivantes:

$$SNR = 10 \log_{10} \frac{P_m}{N_o/2T} \qquad \text{où} \qquad P_m = \frac{A^2}{T} \frac{\left((2N)^2-1\right)}{3}$$

et

$$\sigma^2 = \frac{N_o}{2} (BT)(kT)$$

on obtient:

$$x = P_o \left[\frac{3 \times 10^{SNR/10}}{2\left((2N)^2-1\right)BT \cdot kT}\right]^{\frac{1}{2}}$$

BT est la largeur de bande normalisée et kT est la bande de bruit du filtre de canal. Pour un filtre Butterworth, on a pour kT:

$$kT = \frac{\pi}{2n\sin(\pi/2n)}$$

si n = 5, kT = 1.0166.

Notons en passant qu'il existe un BT optimal où P_e est minimum. Dans le cas du filtre Butterworth d'ordre 5 et d'une "pulse" rectangulaire, on trouve BT = 1.05. Cette valeur reviendra souvent par la suite dans nos résultats. Ajoutons également que cette valeur de BT n'est optimale que pour ce type de filtre et de "pulse". Par exemple, avec un filtre idéal, on obtient BT = 1.0. Il est possible aussi de changer la valeur de l'ordre du filtre qui est fixée à 5 par le paramètre z. Il faut s'assurer à chaque fois de ne pas oublier de modifier la valeur de kT en conséquence. Il faut s'assurer également que la convergence des moments est atteinte à chaque fois. Les essais que nous avons faits nous ont permis de trouver qu'un filtre Butterworth d'ordre >50 était nécessaire pour équivaloir un filtre idéal en ce qui concerne l'IIS.

III.1.3 Calcul des moments d'une interférence sinusoïdale

III.1.3.1 Méthode de calcul

Avec les programmes développés jusqu'ici, pour calculer la probabilité d'erreur en présence d'une interférence sinusoïdale, il nous suffit de connaître les moments de cette interférence.

Supposons que cette interférence à l'entrée du dispositif de décision soit de la forme:

$$I = B\cos\theta \quad \text{où} \quad \theta \text{ est également répartie.}$$

On peut calculer les moments d'ordre 2k de I facilement par:

$$M_I^{2k} = E\{B^{2k}\cos^{2k}\theta\} = \frac{1}{2\pi}\int_{-\pi}^{\pi} B^{2k}\cos^{2k}\theta \, d\theta$$

$$M_I^{2k} = \frac{B^{2k}(2k)!}{2^{2k}(K!)^2} \quad \text{et} \quad M_I^{2k+2} = \frac{B^2}{2}\,\frac{(2k+1)}{(k+1)}\,M_I^{2k}$$

Ce calcul est réalisé dans la procédure MOMBS et le calcul de P_e en présence d'IIS et de cette interférence est réalisé dans le programme QAMBS.

III.1.3.2 Description de QAMBS

Le programme QAMBS, comme on peut le voir en examinant les listings en Annexe et l'organigramme (fig. 3.2), est essentiellement le programme QAM dans lequel on a ajouté la procédure MOMBS. Cette procédure calcule, comme on le sait, les moments de l'interférence sinusoïdale. Le calcul se poursuit ensuite par la combinaison des moments IIS et de moments sinusoïdaux par la procédure UNION. Après cela, le calcul continu comme dans QAM.

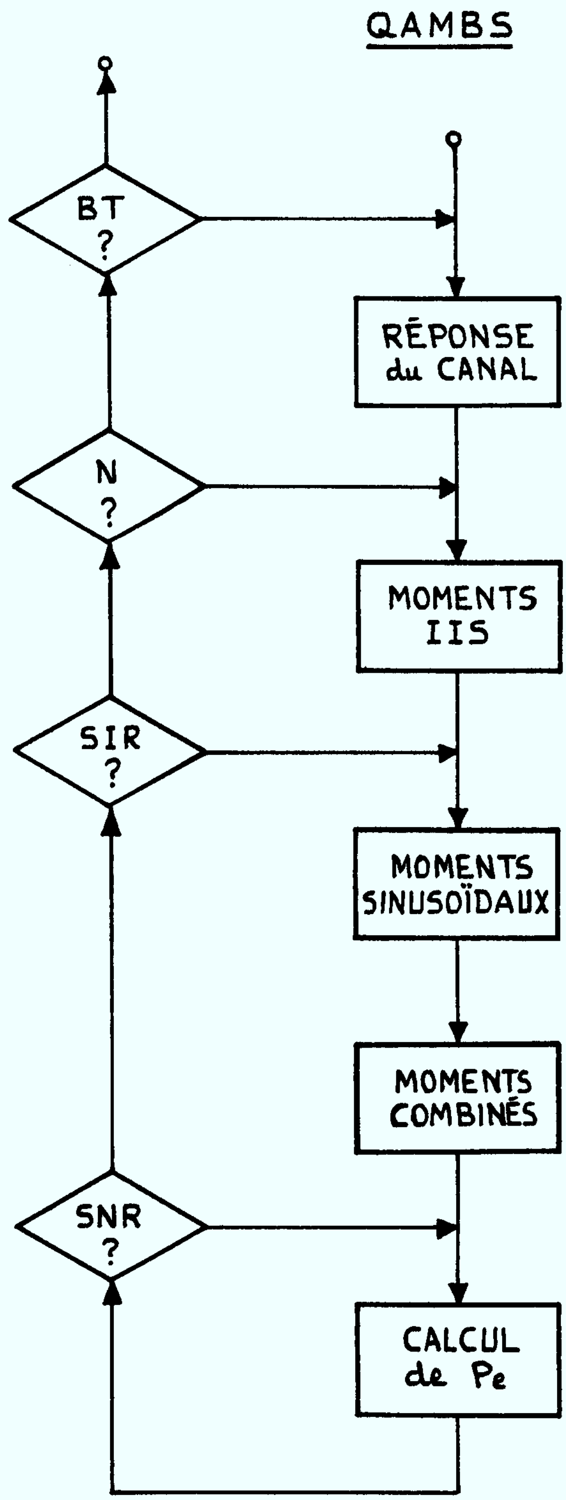


Fig. 3.2 - Organigramme de QAMBS

Dans la procédure MOMBS, le calcul des moments nécessite de connaître B^2. Celui-ci est déterminé à partir de SIR le rapport signal à interférence en dB.

$$SIR = 10 \log_{10} \frac{P_m}{B^2/2T} \quad \text{où} \quad P_m = \frac{A^2}{T} \, \frac{\left((2N)^2-1\right)}{3}$$

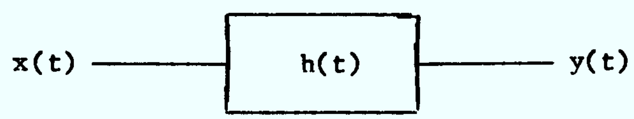
donc

$$B^2 = \frac{2\left((2N)^2-1\right)\cdot 10^{-SIR/10}}{3}$$

III.2 Moments d'une interférence FM

III.2.1 Introduction

Avec comme outil les programmes développés jusqu'à présent, nous nous sommes attardés, comme il se devait, au problème du calcul des moments d'une interférence FM. Avant de décrire la première méthode qui a fait l'objet de nos recherches, nous voudrions souligner la difficulté que représente le calcul d'une manière exacte des moments FM. En fait, la difficulté provient du fait que pour calculer la réponse d'un système linéaire à un processus aléatoire non-gaussien, une connaissance complète des statistiques de tout ordre est en général nécessaire. Supposons le système linéaire suivant:



Si on calcule la statistique d'ordre n de la sortie y(t), on obtient [4]:

$$E\{y(t_1)y(t_2)\ldots y(t_n)\} = E\{x(t_1)x(t_2)\ldots x(t_n)\}*h(t_1)\ldots *h(t_n)$$

où l'opérateur * est l'opérateur de convolution. Cette expression représente une quantité considérable de calcul, rendant toute solution explicite hors de la portée de la plupart des ordinateurs. Quand x(t) est gaussien, la connaissance de $E\{x(t_1)x(t_2)\}$ suffit pour le calcul des statistiques de tout ordre de y(t). De plus, les propriétés d'une fonction de type gaussien permet de prédire le résultat du calcul sans faire aucun calcul. En effet, la convolution d'une fonction gaussienne avec une fonction quelconque donnera une autre fonction gaussienne dont les seuls paramètres à déterminer sont la moyenne et la variance. Il est certain que les propriétés d'un signal FM ne nous permettent pas d'arriver à une telle simplification même si en principe il est possible de calculer les moments de tout ordre d'un signal FM par une transformation non-linéaire.

Il nous apparaît donc nécessaire de faire appel à des hypothèses simplificatrices qui permettront d'obtenir des solutions asymptotiquement valables ou qui représentent une borne d'erreur quelconque.

La première hypothèse que nous avons étudiée est l'hypothèse de quasi-stationnarité. Cette hypothèse est valable asymptotiquement lorsque le signal modulant FM varie très lentement par rapport à la période d'échantillonnage du récepteur. Donc, vu du récepteur, l'interférence FM apparaît comme une interférence sinusoïdale dont la fréquence et l'amplitude sont constantes sur une ou plusieurs périodes d'échantillonnage mais variant lentement selon les statistiques du signal modulant.

La deuxième hypothèse sur laquelle nous avons travaillé est l'hypothèse gaussienne. Cette hypothèse représente le cas limite, ou comme nous allons le voir, la borne supérieure d'une somme très grande d'interférences quelconques indépendantes. En effet, si on découpe la densité spectrale de notre signal FM en un très grand nombre de bandes étroites, on aura une somme d'interférences sinusoïdales dont les amplitudes seront pondérées par la valeur de la densité spectrale à la fréquence de chaque petite bande. Maintenant, si la phase de chacune de ces bandes est indépendante des autres, on aboutit à la limite à une interférence gaussienne.

III.2.2 Hypothèse quasi-stationnaire

III.2.2.1 Méthode de calcul

Cette hypothèse a été utilisée par Morinaga et Namekawa [5] pour le cas de multiples interférences FM dans un récepteur PSK. Toutefois, aucune discussion sur la validité d'une telle hypothèse ne fut présentée.

Donc, si on part avec l'hypothèse de quasi-stationnarité, on aura à l'entrée du dispositif de décision un signal de la forme:

$$s(kT) = KG(f_i)\cos\theta$$

où G(f) représente la réponse en fréquence du récepteur et θ est répartie également. La variable aléatoire f_i représente la fréquence instantanée basse-fréquence du signal FM dont les statistiques dépendent du signal modulant. Nous avons choisi pour f_i une distribution gaussienne. Ce choix demeure un choix réaliste et permet de simplifier les calculs. De plus, on peut choisir G(f) pour simplifier les calculs davantage. Le cas le plus simple est le cas où G(f) est un filtre idéal. Dans ce cas, la variable aléatoire $G(f_i)$ devient une variable binaire prenant les valeurs 1 ou 0 dépendant si f_i est à l'intérieur ou non de la bande passante du récepteur. La procédure qui suivra est alors très simple:

1) on calcule P_e sans interférence (P_1),
2) on calcule P_e avec interférence sinusoïdale $I = K\cos\theta : (P_2)$,
3) on fait la moyenne pondérée de ces deux valeurs.

Si $P_k = P_r(\left|f_i\right| < B/2)$, alors on aura:

$$P_e = p_k P_2 + (1-p_k)p_1 = P_1 + P_k(p_2-p_1) \qquad p_2 > P_1$$

On calcule P_k par:

$$P_k = \frac{1}{2}\left[\operatorname{erfc}\left(\frac{\tilde{f}+B/2}{\sqrt{2}\ \Delta f}\right) - \operatorname{erfc}\left(\frac{\tilde{f}-B/2}{\sqrt{2}\ \Delta f}\right)\right]$$

où $\tilde{f}$: écart de fréquence entre les porteuses FM et QAM,
Δf : déviation de fréquence RMS du signal FM.

Pour les fins de calculs, $\tilde{f}$ et Δf sont normalisés à B/2 et

$$Y = \frac{\Delta f}{B/2} \qquad X = \frac{\tilde{f}}{B/2}$$

III.2.2.2 Difficultés numériques

Comme on peut le voir, le calcul qui en résulte est très simple. Par contre, l'emploi d'un filtre idéal bien que facilitant le calcul de l'interférence FM complique le calcul de l'IIS. En effet, il faudra vérifier que la convergence des moments IIS est bien atteinte pour ce nouveau type de filtre. En fait, la décroissance des échantillons de la réponse du canal est très lente et la convergence n'est possible, en pratique, avec un nombre pas trop élevé d'échantillons, que pour BT = 1.0. De plus, pour ce type de filtre et une "pulse" rectangulaire, on obtient un niveau d'interférence IIS relativement élevé par rapport à une situation réelle (voir fig. 3.3). On a pu vérifier en calculant P_e pour différents ordres de filtre Butterworth que lorsque l'on fait tendre l'ordre du filtre vers l'infini, l'IIS croît d'une manière constante et atteint son maximum pour un filtre idéal (voir fig. 3.4).

Nos efforts subséquents ont tenté de réduire ce niveau d'IIS tout en couvrant notre filtre idéal. Nos essais ont porté surtout sur le type de "pulse" à la transmission. On s'attendait à priori que l'emploi d'une "pulse" du type cosinus relevé viendrait diminuer l'IIS par rapport à une "pulse" rectangulaire. En fait, les résultats ont démontré le contraire, l'IIS étant plus élevé pour le cas de la "pulse" cosinus relevé. On aurait pu prévoir le résultat en examinant l'allure du spectre de chacune de ces "pulses".

Dans les deux cas, on a un spectre de forme similaire (voir fig. 3.5). On s'attend à ce que l'IIS soit moindre si H(BT/2) est près de l'unité. Si $H(\Omega)$ est constant dans la bande BT, l'IIS sera nul. On peut vérifier que:

$H(BT/2) = 0.64$	"pulse" rectangulaire
$H(BT/2) = 0.50$	"pulse" cosinus relevé avec recouvrement sur 2 intervalles.

Donc, l'IIs sera plus élevé pour une "pulse" cosinus relevé que pour une "pulse" rectangulaire. Ce résultat ne vaut évidemment que pour le filtrage idéal.

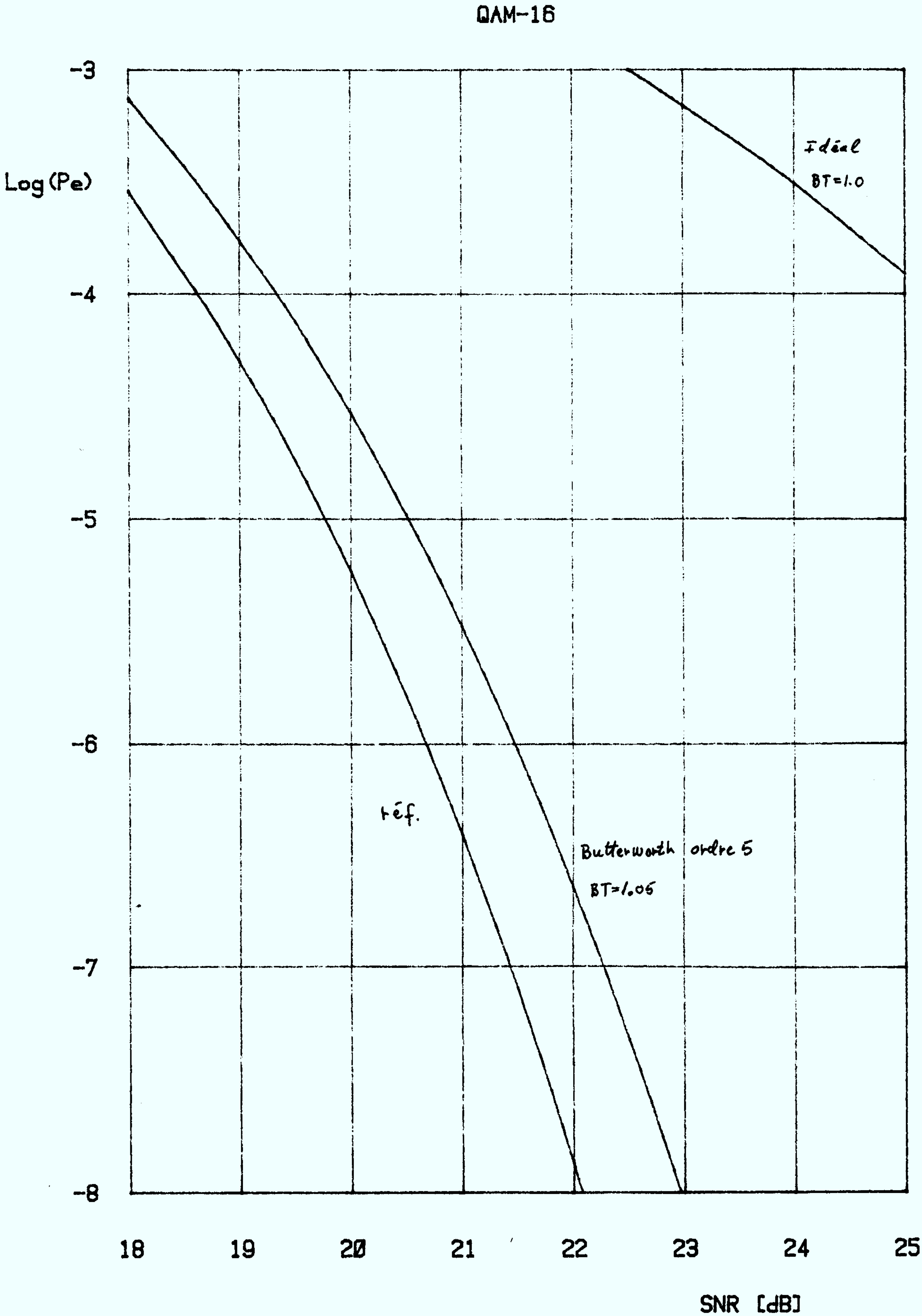

QAM-16
Log(Pe)
-3
-4
-5
-6
-7
-8
18
19
20
21
22
23
24
25
SNR [dB]
Idéal
BT=1.0
réf.
Butterworth ordre 5
BT=1.05

FIGURE 3.3

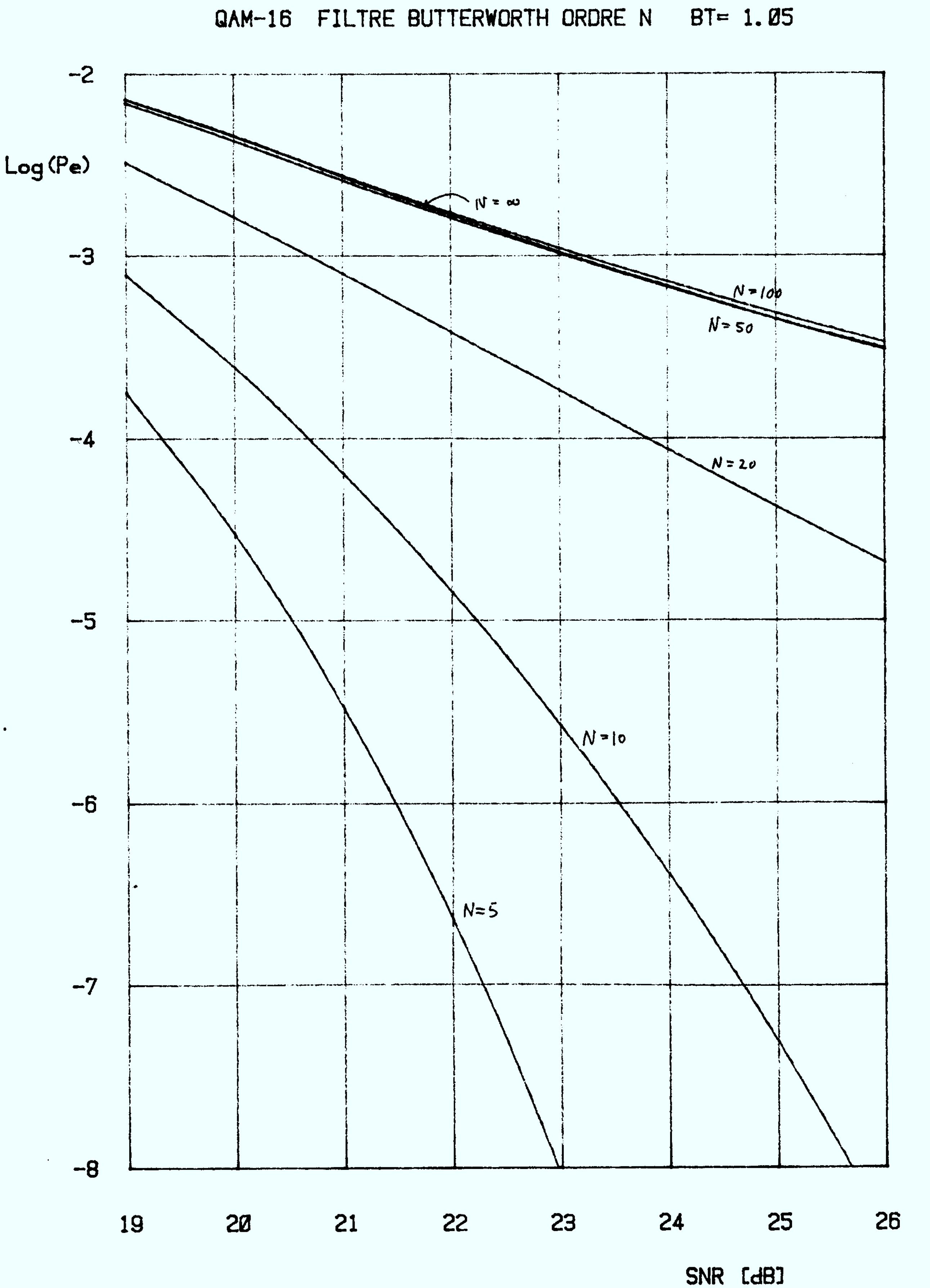

QAM-16 FILTRE BUTTERWORTH ORDRE N BT= 1.05
Log(Pe)
-2
-3
-4
-5
-6
-7
-8
N = ∞
N = 100
N = 50
N = 20
N = 10
N = 5
19
20
21
22
23
24
25
26
SNR [dB]

FIGURE 3.4

III.2.2.3 Difficultés théoriques

L'emploi d'un filtre idéal rend l'hypothèse quasi-stationnaire encore plus invraisemblable, comme on le verra en étudiant la borne de Weiner [6].

Soit y(t) un signal FM à la sortie d'un filtre linéaire réel dont la réponse en fréquence est rationnelle.

$$y(t) = v(t) + v_c(t)$$

où v(t) est le terme quasi-stationnaire et $v_c(t)$ un terme d'erreur que l'on désire le plus faible possible. Dans son article, Weiner donne la borne suivante:

$$|v_c(t)| < \Delta\omega |f'(\tau)|_{max} \sum_{y=1}^{n} \frac{|K_y|}{(\alpha_y)^3} = \Delta\omega |f'(\tau)|_{max} K(n)$$

comme H(p) est rationnel, $H(p) = \sum_{y=1}^{n} \frac{K_y}{p-p_y} \qquad p_y = \alpha_y + j\beta_y$

On peut calculer K(n) pour un filtre Butterworth. Le résultat est présenté au tableau 3.1. Comme on peut le constater, K(n) croît avec n et tend vers l'infini pour un filtre idéal. Il devient donc impossible de trouver des valeurs raisonnables de $\Delta\omega |f'(\tau)|_{max}$ qui puissent rendre $|v_c(t)|$ aussi petit que l'on veut.

III.2.2.4 Résultats

Pour atténuer les difficultés rencontrées, tant numériques que théoriques, nous avons préféré présenter des résultats avec un filtre Butterworth d'ordre 5. Avec ce type de filtre, le calcul des moments FM n'est plus aussi simple car il faut calculer les moments de G(f) par une transformation non-linéaire. Mais nous pensons qu'on peut éviter ces complications en considérant le problème autrement. Supposons la bande de notre filtre Butterworth divisée en trois zones disjointes (voir fig. 3.6). La zone 1 et la zone 3 représentent le cas précédent du filtrage idéal car la valeur de la réponse est 1 ou 0. Dans la zone de transition (ou zone 2), la réponse peut prendre

n	K(n)
1	1
2	4
3	10.24
4	20.91
5	37.40
10	311.20
15	2,324.32
20	26,397.98
25	385,365.80
30	6,143,691.96

Tableau 3.1

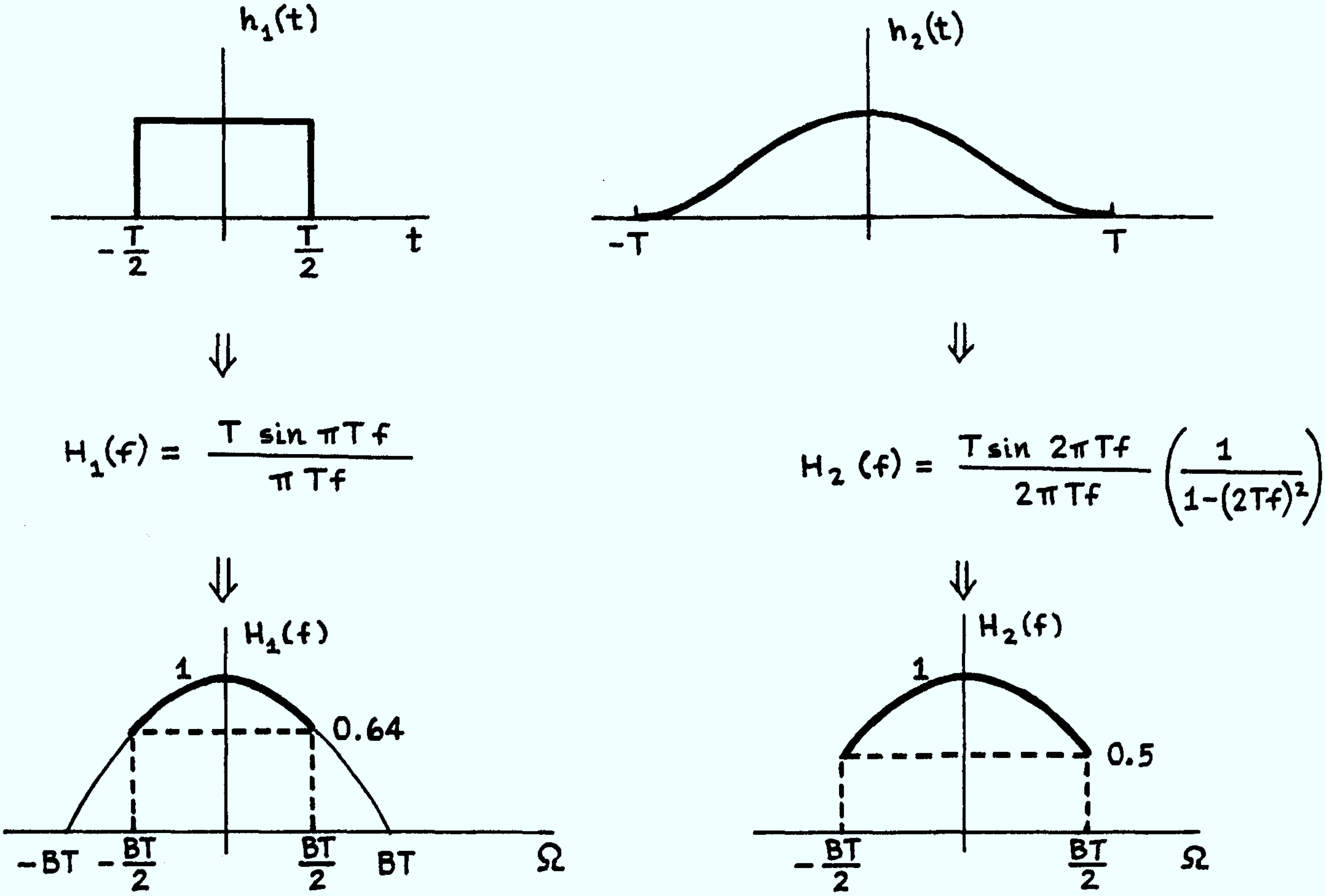


Fig. 3.5 - Comparaison de deux formes de "pulse" différentes

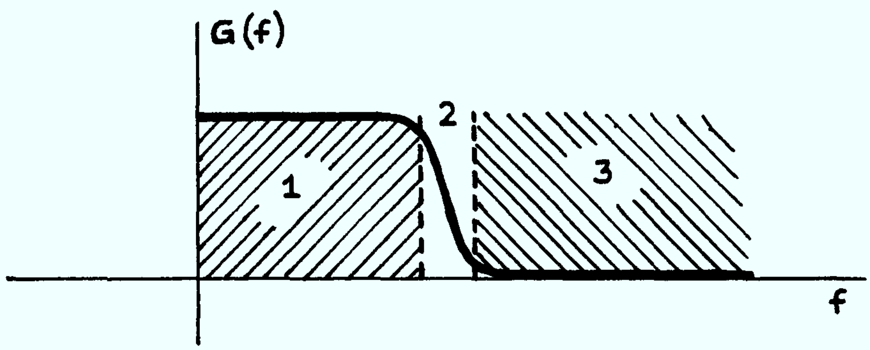


Fig. 3.6 - Modèle de filtrage pour les besoins du calcul de l'interférence FM

une valeur continue entre 0 et 1. On pourrait assigner à cette zone la valeur 1/2 et faire la moyenne sur les trois valeurs de P_e correspondantes. Toutefois, comme la probabilité que la fréquence instantanée soit dans cette zone est faible, puisque cette zone est généralement étroite, on peut tout simplement négliger cette zone et faire le calcul comme dans le cas idéal sans commettre d'erreurs importantes.

Donc, les résultats présentés en annexe A valent pour un filtre Butterworth ordre 5 avec BT = 1.05 et pour le QAM-16. Sur ces courbes, le paramètre Y correspond à la déviation de fréquence RMS normalisée à B/2 et la position FM correspond au paramètre X qui est l'écart entre les porteurs FM et QAM normalisés à B/2. Le paramètre SIR a déjà été défini précédemment.

Le premier type de courbes présente Log(P_e) en fonction de X pour différents SIR et Y. Le deuxième type de courbes présente le DSNR (accroissement de SNR pour garder un P_e constant) en fonction de SIR pour un $P_e = 10^{-5}$ avec et sans IIS pour deux valeurs de Y. Ces courbes sont loin de former un ensemble complet mais elles permettent d'avoir une idée suffisante de l'allure des résultats.

Ces résultats ont été réalisés avec le programme QAMFM. Il est essentiellement semblable à QAMBS avec en plus le calcul de P_k. L'organigramme de QAMFM est présenté à la figure 3.7. Ce programme imprime, en plus de la valeur des différents paramètres, six valeurs différentes de probabilité d'erreur. Elles correspondent aux situations suivantes:

P_{E1} : P_e avec bruit gaussien seulement.

P_{E2} : P_e avec bruit gaussien et interférence sinusoïdale constante.

P_{E3} : comme P_{E2} mais avec en plus l'IIS.

P_{E4} : comme P_{E1} mais avec IIS.

P_{E5} : P_e avec interférence FM avec IIS.

P_{E6} : P_e avec interférence FM sans IIS.

Les valeurs tracées sur les courbes correspondent à P_{E5}.

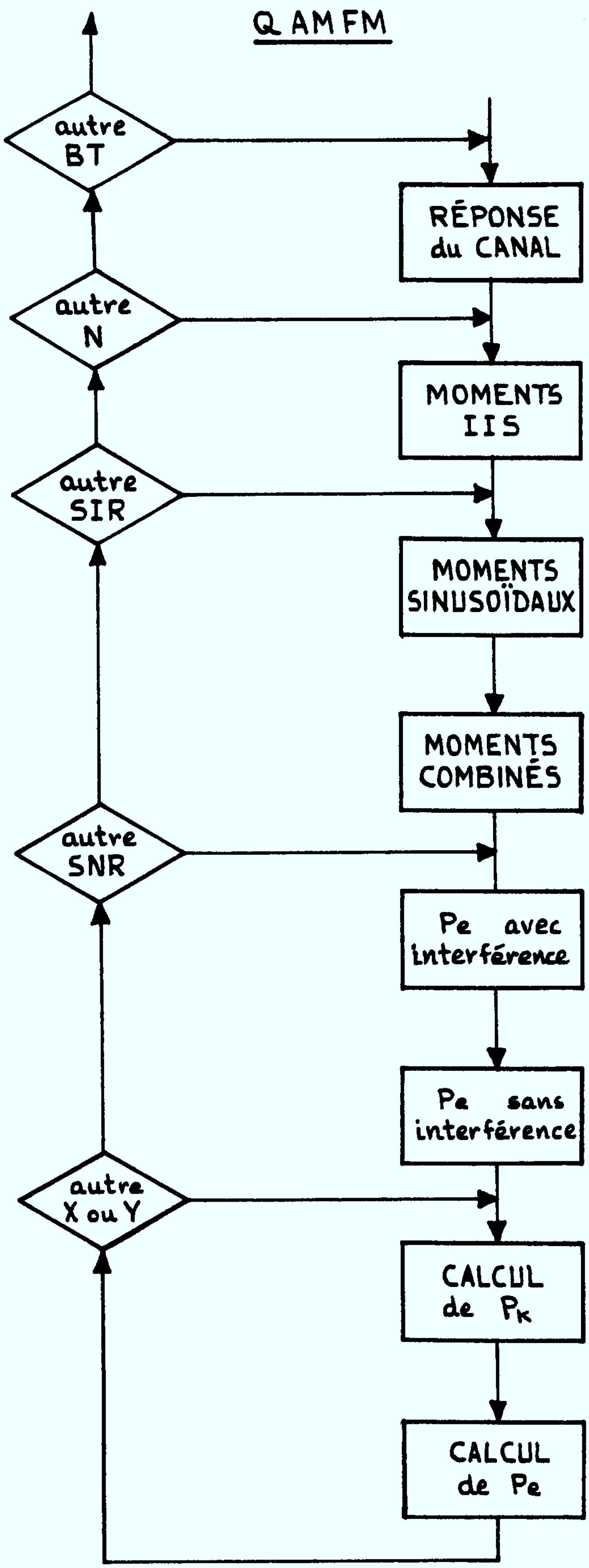


Fig. 3.7 - Organigramme de QAMFM

III.2.3 Hypothèse gaussienne

III.2.3.1 Introduction

Soit G(f) la densité spectrale de notre interférence FM. Nous voulons découper la bande de cette densité spectrale en n bandes disjointes. À la limite, lorsque n est grand, chacune de ces petites bandes pourrait être vue comme une interférence sinusoïdale de la forme:

$$I_i = B_i \cos\theta_i$$

où l'indice i représente la bande i située à la fréquence f_i, $B_i = \sqrt{G(f_i)}$ et θ_i est répartie également. Donc, notre interférence FM sera constituée par la somme de n interférences sinusoïdales qui, si on suppose les θ_i indépendants, aura une distribution gaussienne lorsque n tendra vers l'infini. Donc, l'hypothèse gaussienne constitue le cas limite d'une somme infinie d'interférences quelconques indépendantes.

III.2.3.2 Méthode de calcul

On s'est d'abord intéressé au cas d'un nombre fini d'interférence sinusoïdale pour G(f) blanc et G(f) de forme gaussienne. Pour ce faire, la méthode des moments nous a été très utile. Toutefois, les temps de calculs deviennent très élevés lorsque le nombre de bandes augmente substentiellement. Ceci est dû au fait qu'il faut combiner les moments de chaque interférence ce qui demande passablement de temps de calcul. Cependant, ceci nous a permis de constater le comportement de P_e lorsque n augmente. En examinant la figure 3.8, on peut voir que P_e augmente de manière monotone avec n. Donc, l'hypothèse gaussienne qui est le cas limite lorsque n est infini représente une borne supérieure pour une somme d'interférences indépendantes. On observe le même comportement peu importe la forme de G(f). Donc, pour calculer les moments de notre interférence gaussienne, il suffit de calculer la variance qui représente en fait la puissance de notre interférence dans la bande du récepteur. Ensuite, on peut procéder au calcul de P_e par la méthode des moments. En fait, le calcul par la méthode des moments n'est plus nécessaire puisqu'on peut assimiler notre interférence au bruit gaussien. Donc, finalement, le calcul se résume à quelques opérations simples:

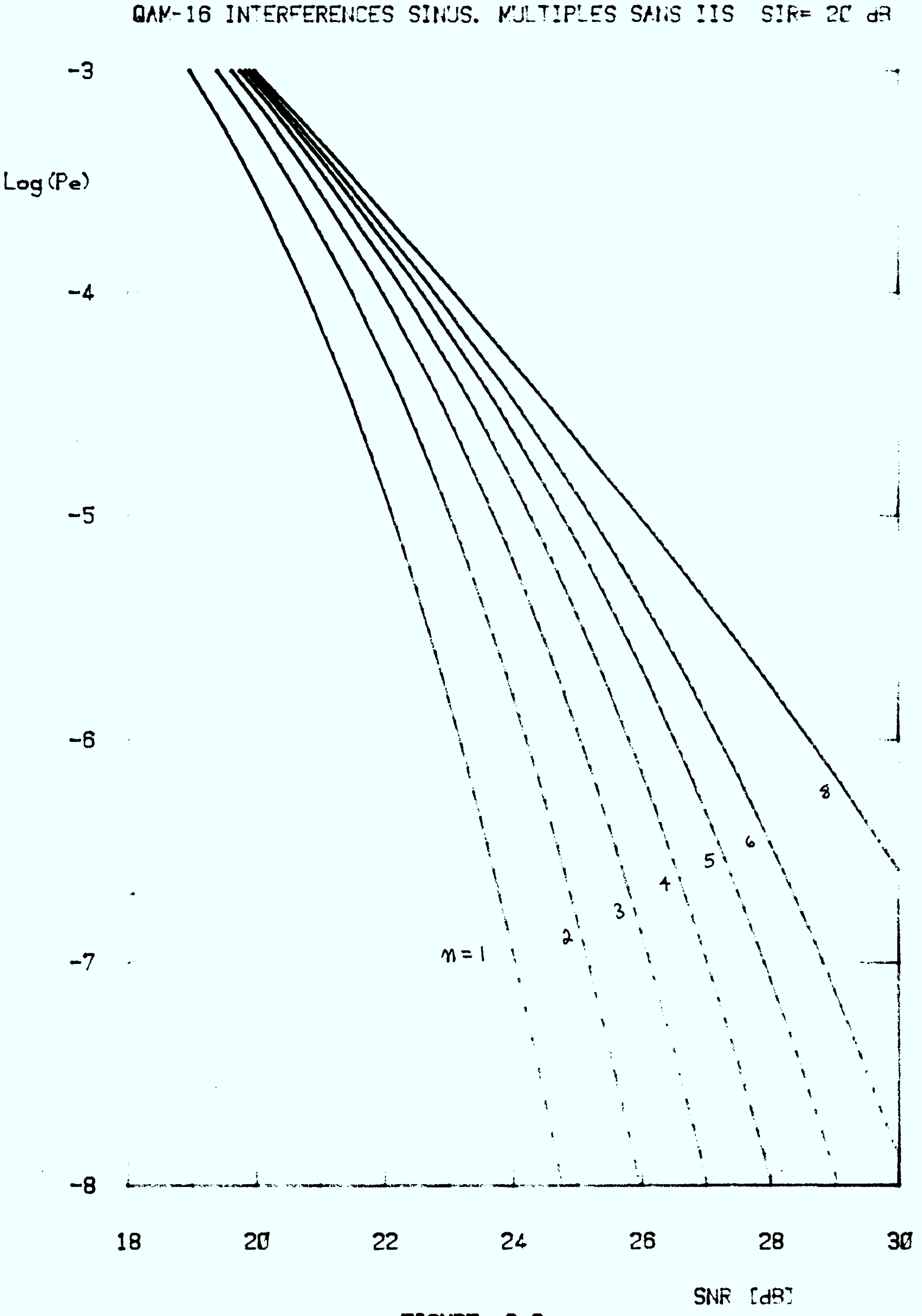

QAM-16 INTERFERENCES SINUS. MULTIPLES SANS IIS SIR= 20 dB
Log (Pe)
-3
-4
-5
-6
-7
-8
n = 1
2
3
4
5
6
8
18
20
22
24
26
28
30
SNR [dB]

FIGURE 3.8

1) Calculer la puissance de l'interférence à l'entrée du dispositif de décision en intégrant la densité spectrale dans la bande du récepteur. Dans nos résultats, ceci a été réalisé avec une densité spectrale de forme gaussienne, ce qui simplifie passablement le calcul.

2) On recalcule un nouveau SIR appelé SIRC:

$$SIR = 10 \log_{10} \frac{P_m}{P_I}$$

où P_m = puissance moyenne du signal QAM,
P_I = puissance moyenne de l'interférence à l'entrée du récepteur.

Si on a $P = \int_{-\infty}^{\infty} G(f)\left|H(f)\right|^2 df$ % de la puissance de l'interférence qui entre dans le récepteur.

où $H(f)$ est la réponse du récepteur $\left(H(f)\text{ est idéal dans nos résultats}\right)$ et $G(f)$ est normalisé, on aura:

$$SIRC = 10 \log_{10} \frac{P_m}{P \cdot P_I} = SIR - 10 \log_{10} P$$

3) On recalcule un nouveau SNR appelé SNIR pour rapport signal à bruit + interférence:

$$SNIR = 10 \log_{10} \frac{S}{N+I} = 10 \log_{10}\left[10^{-SNR/10} + 10^{-SIRC/10}\right]^{-1}$$

4) On calcule P_e par:

$$P_e = \frac{1}{2}(2-1/N)\operatorname{erfc}(S)$$

où

$$S = \left(\frac{3\times10^{SNIR/10}}{2\left((2N)^2-1\right)}\right)^{\frac{1}{2}}$$

Tout ce calcul peut être réalisé sur un petit calculateur. La seule difficulté numérique étant le calcul d'une intégrale numérique pour le calcul de erfc(x) et de P. Ceci a été réalisé dans le programme OFMRG (voir fig. 3.9). Évidemment, tout ceci est très simple car on néglige l'IIS. Pour l'inclure, on aurait simplement qu'à revenir à la méthode des moments, ce qui ne devrait pas poser de problèmes.

III.2.3.3 Description de QFMRG

Ce programme fournit, pour fin de comparaison, les deux valeurs de P_e correspondant à la situation avec ou sans interférences. Le programme permet d'utiliser un G(f) quelconque. Il suffit de modifier la fonction FT dans la fonction BANDE. Les deux paramètres relatifs à l'interférence FM sont DF et FC. Ces deux paramètres sont normalisés à B/2 comme auparavant.

FC a la même signification que dans l'hypothèse quasi-stationnaire; c'est l'écart entre les porteuses FM et QAM normalisé à B/2. Par contre, DF a une signification différente. Ici, il représente l'écart type de G(f) normalisé à B/2 alors qu'avant, le paramètre Y représentait la déviation de fréquence RMS du signal FM qui est en fait l'écart type du signal modulant normalisé à B/2.

Des résultats sont présentés en annexe B pour le OAM-4, le OAM-16 et le OAM- 64. Pour chacun de ces cas, on fournit une série de courbes de P_e en fonction de SNR pour différentes valeurs de DF, FC et SIR. Une courbe supplémentaire est présentée pour le cas d'une interférence centrée à faible largeur de bande (FC = 0 et DF = 0.1) pour différentes valeurs de SIR. Tous ces résultats valent pour un filtre idéal et sans IIS.

III.3 Discussion

III.3.1 Présentation générale

Prenons d'abord les résultats présentés en annexe 2 concernant l'hypothèse quasi-stationnaire. On remarque d'abord que ceux-ci sont très partiels et ne valent que pour le QAM-16. Ceci contraste avec les résultats en annexe B concernant l'hypothèse gaussienne. Pour ces résultats, on fournit un ensemble de courbes assez complet pour le QAM-4, le OAM-16 et le QAM-64. D'abord,

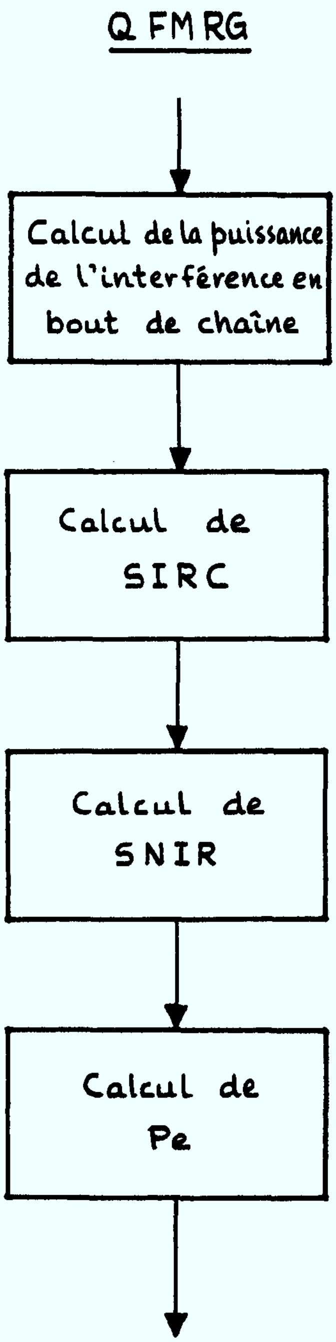


Fig. 3.9 - Organigramme de QFMRG

nous pensons que les résultats en annexe 3 sont plus réalistes et de plus, ils sont plus faciles à obtenir, les calculs étant réalisables sur un ordinateur de petite taille.

D'autres différences de présentations peuvent être notées comme par exemple l'échelle horizontale des courbes. Dans le premier cas, on présente $\log(P_e)$ en fonction de la position FM (paramètre X) pour différentes valeurs de SNR. Dans l'autre cas, on présente $\log(P_e)$ en fonction de SNR pour différentes valeurs de FC. Nous pensons que la première présentation est préférable à la seconde car elle permet, tout de suite, de mieux voir la variation de P_e en fonction d'un paramètre de l'interférence FM. Par contre, dans le deuxième cas, on a conservé le paramètre SNR sur l'échelle horizontale en prévision du calcul d'une borne supérieure. En fait, cette borne nous permet de déterminer dans quelle plage de SNR, l'hypothèse gaussienne serait une borne supérieure absolue.

Un autre point est à remarquer en annexe 2 en ce qui concerne les courbes; celles-ci sont présentées en tenant compte de l'IIS alors que celles en annexe 3 n'en tiennent pas compte. Il est certain que si on désire comparer les courbes entre elles, il faudra connaître à priori la valeur de P_e à soustraire due à l'IIS seulement. En examinant ces courbes, quelque soit l'hypothèse utilisée, on remarque un comportement semblable. En effet, P_e est inversement proportionnel à SNR et à SIR, ce qui n'est pas surprenant. On remarque aussi que P_e diminue d'une manière monotone à mesure que l'écart entre les porteuses QAM et FM augmente et cela pour une dispersion spectrale FM donnée. Ce résultat va de soi également. De plus, on remarque que pour un écart fixe entre les porteuses, P_e peut varier dans un sens ou dans l'autre en fonction de la dispersion du spectre FM (spectre large ou étroit). Si l'écart est faible, un spectre FM étroit causera plus de dégradation au QAM qu'un spectre large. Par contre, si l'écart est grand, on observe le phénomène inverse. Par exemple, prenons les courbes des figures A.2.1 et A.2.2 pour SNR = 30 dB. Pour X entre 0 et 1.5, la valeur de P_e obtenue pour Y = 1 est supérieure à celle obtenue pour Y = 2. Par contre, entre X = 1.5 et X = 5, c'est l'inverse. Ce comportement est général et se retrouve également dans les résultats de l'annexe 3. On peut comprendre cela en sachant que dans les deux cas, P_e est fonction uniquement de la puissance totale de l'interférence entrant dans le récepteur QAM. Cela implique que pour une interférence centrée, où toute la puissance passe dans le récepteur, P_e est indépendant de la densité spectrale de l'interférence. Cette dernière observation nous laisse entrevoir une certaine imprécision des deux hypothèses utilisées. En effet, nous pensons qu'en réalité, une interférence à large bande cause plus de dégradation de P_e qu'une interférence à bande étroite pour la même puissance totale entrant dans le récepteur mais nos deux approches ne permettent pas de rendre compte de ce phénomène.

III.3.2 Comparaisons

Pour pouvoir mieux comparer les résultats obtenus par les deux hypothèses, nous avons tracé à la figure 3.10, P_e pour le QAM-16 sans IIS avec SIR = 20 dB en fonction de SNR pour le cas où toute la puissance de l'interférence entre dans le récepteur. Si on s'y réfère, on y voit trois courbes numérotées 1, 2 et 3. La courbe #1 est tout simplement le P_e obtenu avec bruit de canal gaussien seulement. La courbe #2 représente le P_e obtenu avec une interférence sinusoïdale constante. Toutes les courbes obtenues avec l'hypothèse quasi-stationnaire pour différentes valeurs de X et Y, seront situées quelque part entre les courbes 1 et 2. La courbe #3 présente le P_e obtenu par l'hypothèse gaussienne et toutes les courbes obtenues sous cette hypothèse pour différentes valeurs de FC et DF seront situées entre les courbes 1 et 3.

On voit donc très nettement l'écart considérable qui existe entre les courbes 2 et 3 donc entre les 2 hypothèses, l'hypothèse quasi-stationaire étant nettement plus conservatrice. C'est d'ailleurs pour cette raison, qu'après avoir obtenu comme premier résultat, les résultats de l'hypothèse quasi-stationnaire, nous avons cherché une autre méthode donnant des résultats plus réalistes. Sous l'hypothèse gaussienne, nous pensons être dans un cas intermédiaire entre le pire cas et l'hypothèse quasi-stationnaire. En effet, l'hypothèse gaussienne présuppose indépendance des phases entre les différentes bandes spectrales. Cette condition n'est certes pas le pire cas, ni le meilleur. Le pire cas serait que toutes les différentes composantes de l'interférence seraient toujours en phase, ce qui n'est pas très réaliste. Donc, nous pensons que l'hypothèse gaussienne pourrait fournir des résultats valables si on les compare aux résultats de l'hypothèse quasi-stationnaire qui sont nettement trop conservateurs et le pire cas qui ne serait pas réaliste du tout.

Un autre point est à observer concernant la figure 3.10. On remarque que la courbe #2 semble toujours chevaucher la courbe #1 quelque soit SNR tandis que la courbe #3 semble tendre vers une valeur limite de P_e quand SNR tend vers l'infini. Ce dernier comportement s'explique facilement car quand le SNR est faible c'est le bruit de canal qui prédomine tandis que lorsque SNR est grand c'est l'interférence. À la limite, pour un SIR = 20 dB, quand SNR tend vers l'infini, P_e devrait tendre vers la valeur qu'on obtiendrait sur la courbe #1 avec SNR = 20 dB. Pour ce qui est de la courbe #2, on n'observe pas du tout le même comportement, probablement parce que l'équivalence en terme de P_e, entre une interférence sinusoïdale et du bruit gaussien, se fait à un niveau de SNR très élevé ou à un P_e très faible pour un SIR = 20 dB.

III.3.3 Borne supérieure

Nous avons remarqué précédemment que l'hypothèse gaussienne était plus réaliste que l'hypothèse quasi-stationnaire sans être en mesure de dire jusqu'à quel point on est près de la réalité. Pour combler cette lacune, nous avons pensé faire le calcul d'une borne supérieure pour P_e qui nous permettrait de voir si l'hypothèse gaussienne dans une certaine plage de SNR pouvait être considérée comme une borne supérieure absolue pour le cas d'interférence non-gaussienne quelconque. Plusieurs méthodes existent dans la littérature et ce calcul ne devrait pas, en principe, poser de problèmes majeurs. Toutefois, les contraintes de temps ne nous ont pas permis de le réaliser. Nous avons donc remis ce travail dans le cadre de travaux futurs.

QAM-16 SANS IIS SIR= 20 dB

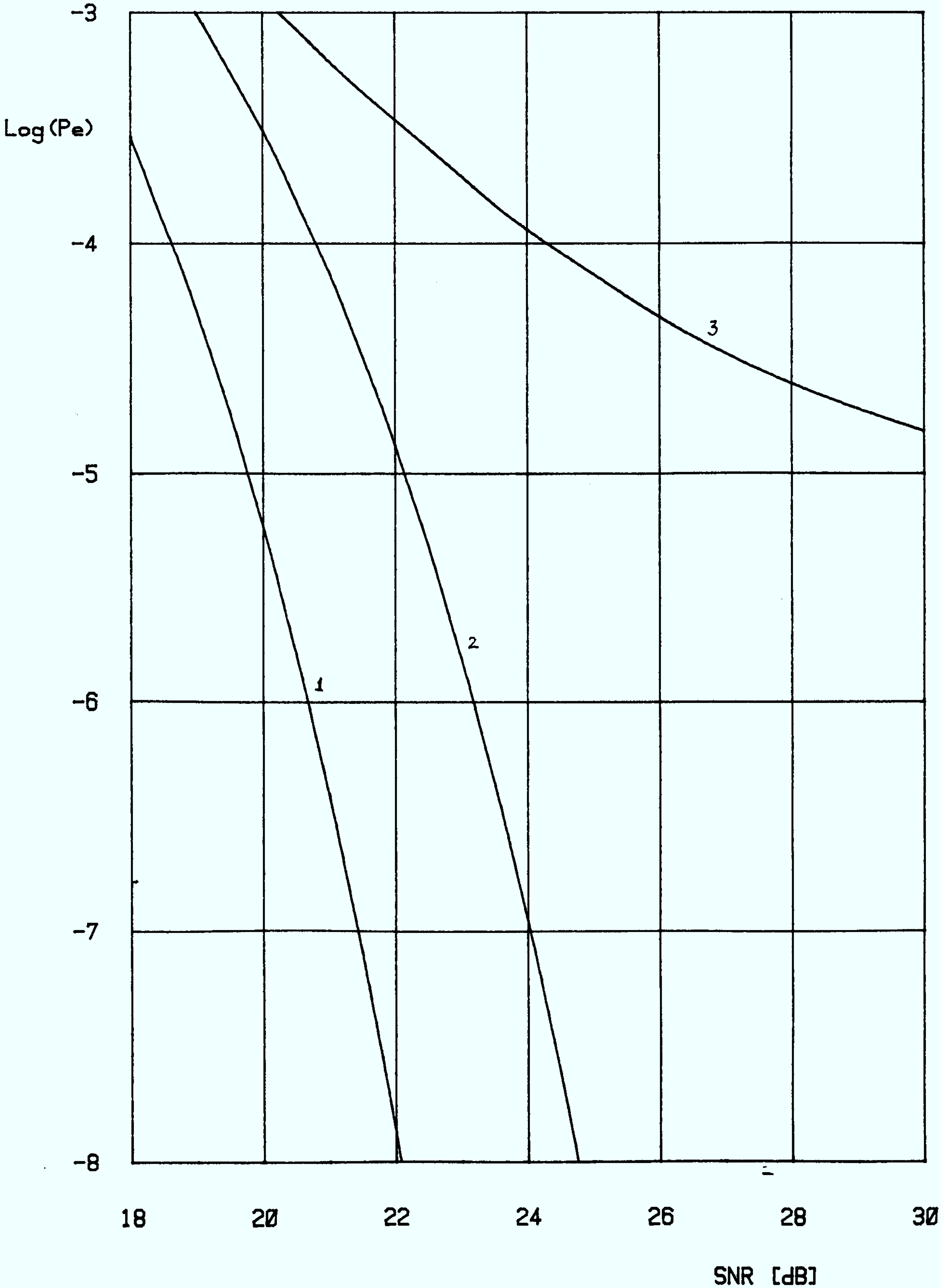


FIGURE 3.10

CHAPITRE IV

CONCLUSION

Nous présentons dans ce travail une méthologie flexible pour étudier la performance des récepteurs QAM face aux interférences de toute nature dont les signaux FM des canaux voisins.

Cette méthodologie est basée sur une idée simple proposée par Ho et Yeh. Il s'agit tout simplement d'exprimer la probabilité d'erreur en fonction des moments de la variable représentant la somme de toutes les interférences présentes à l'entrée du récepteur. Comme ces interférences sont indépendantes, il suffit de calculer les moments de chacune d'elles séparément, et de les combiner par la suite suivant les règles précises.

La composante d'interférence, qui nous intéresse le plus, est un signal FM à très grande largeur de bande. S'appuyant sur l'hypothèse de "quasi-stationnaritę", son comportement sur des intervalles de temps faible est équivalent à un signal sinusoïdal dont la fréquence correspond à la fréquence instantannée du signal FM. Cette hypothèse simplificatrice permet de traiter, de manière simple, l'interférence due à un signal FM.

Nous avons également proposé une autre hypothèse basée sur la notion "Equivalence de puissance". Cette hypothèse se justifie physiquement par la contrainte imposée sur la puissance totale dans la bande passante QAM. Toutefois, on s'attendait à ce que cette hypothèse aboutisse à des résultats plus pessimistes que ceux obtenus par l'hypothèse "quasi-stationnarité".

Dû à notre choix de formulation mathématique, un filtre de Butterworth a été choisi comme modèle du canal de transmission à la place d'un canal de

Nyquist. Ce choix introduit automatiquement de l'interférence intersymbole à la sortie du récepteur. Par ailleurs, ce facteur a été minimisé par un choix judicieux des paramètres en jeu tels que: l'instant d'échantillonnage, la largeur de bande et l'ordre du filtre dont la phase a été linéairement égalisée.

Les résultats obtenus sont finalement présentés sous forme de famille de courbes normalisées. Ils sont donc utilisables pour toutes situations quelconques. Il est intéressant de noter que ces résultats apparaissent à nos yeux comme non complets. En effet, on a obtenu des résultats pour l'interférence que provoque un signal FM. Cependant, pour les obtenir numériquement, nous avons dû utiliser les deux hypothèses: l'une très optimiste, l'autre très pessimiste. Il reste à savoir si ces résultats se trouvent à l'intérieur des bornes pessimistes et optimistes générales. Le calcul de ces bornes sera effectué dans le prochain contrat, dans lequel on propose d'analyser l'interférence FM à bande étroite.

Un dernier point, non moins important à souligner, est la complexité de la méthode utilisée. D'une part, elle consomme beaucoup de temps de calcul; ce phénomène ne représente plus un vrai désavantage, avec l'avènement des mini et micro-ordinateurs. D'autre part, cette complexité a nécessité un effort très sérieux (une année-ingénieur) pour réussir sa mise en oeuvre. Le programme, écrit en PASCAL, est modulaire et flexible, permettant aux usagers de varier tous les paramètres qu'ils désirent, tout en assurant un fonctionnement relativement raisonnable. Ce coût du développement semble à prime abord assez élevé. Toutefois, il est maintenant disponible de manière quasi-publique. Il ne faut pas oublier qu'avant l'octroi du contrat donnant naissance à ce rapport, les programmes de cette nature étaient des propriétés privées jalousement gardées par les compagnies telles que BNR, AT & T, ERICSON, etc.. C'est donc un grand pas de franchi; il ne nous reste qu'à souhaiter une collaboration aussi fructueuse entre les chercheurs universitaire et les organismes publiques.

Bibliographie

[1] Ho et Yeh, "Error probability of a multilevel digital system with intersymbol interference and gaussian noise", Bell System Technical Journal, U.S.A., March 1973, 50, n° 3, pp. 1017-1023.

[2] H.T. Huynh, P. Fortier, "Étude des défauts systématiques et naturels dans un réseau à micro-onde, Phase I: Synthèse bibliographique", Université Laval, Québec.

[3] P. Fortier, "Analyse des performances des modulations d'amplitude en quadrature à haute capacité", thèse de maîtrise, Université Laval, avril 1984.

[4] A. Papoulis, "Probability, random variables and stochastic process", McGraw-Hill, 2è édition, 1984.

[5] N. Morinaga, T. Namekawa, "Intersystem interference between analog and digital systems", IEEE International Conference on Communication, June 1982, Philadelphia, U.S.A.

[6] D.D. Weiner, B.J. Leon, "The Quasi-stationnary response of linear system to modulated waveform", Proc. IEEE, Vol. 53, N° 6, June 1965.

[7] S. Kabasawa et al., "M-ary CPSK Detection with noisy reference and interferences", IEEE Trans. on AES, Vol. AES-16, N° 5, September 1980.

[8] M Abramowitz et I.A. Stegun, "Handbook of mathematical functions", National Bureau of Standards, December 1965.

[9] P. Dupuis et al., "16-QAM Modulation for high capacity digital radio system", IEEE Trans. Com., Vol. COM-27, December 1979.

[10] J.G. Proakis, "Digital Communications", McGraw-Hill 1983.

ANNEXE 1

EXPRESSION DE LA PROBABILITÉ D'ERREUR

Considérons la figure A.1 représentant la constellation QAM à 16 états.

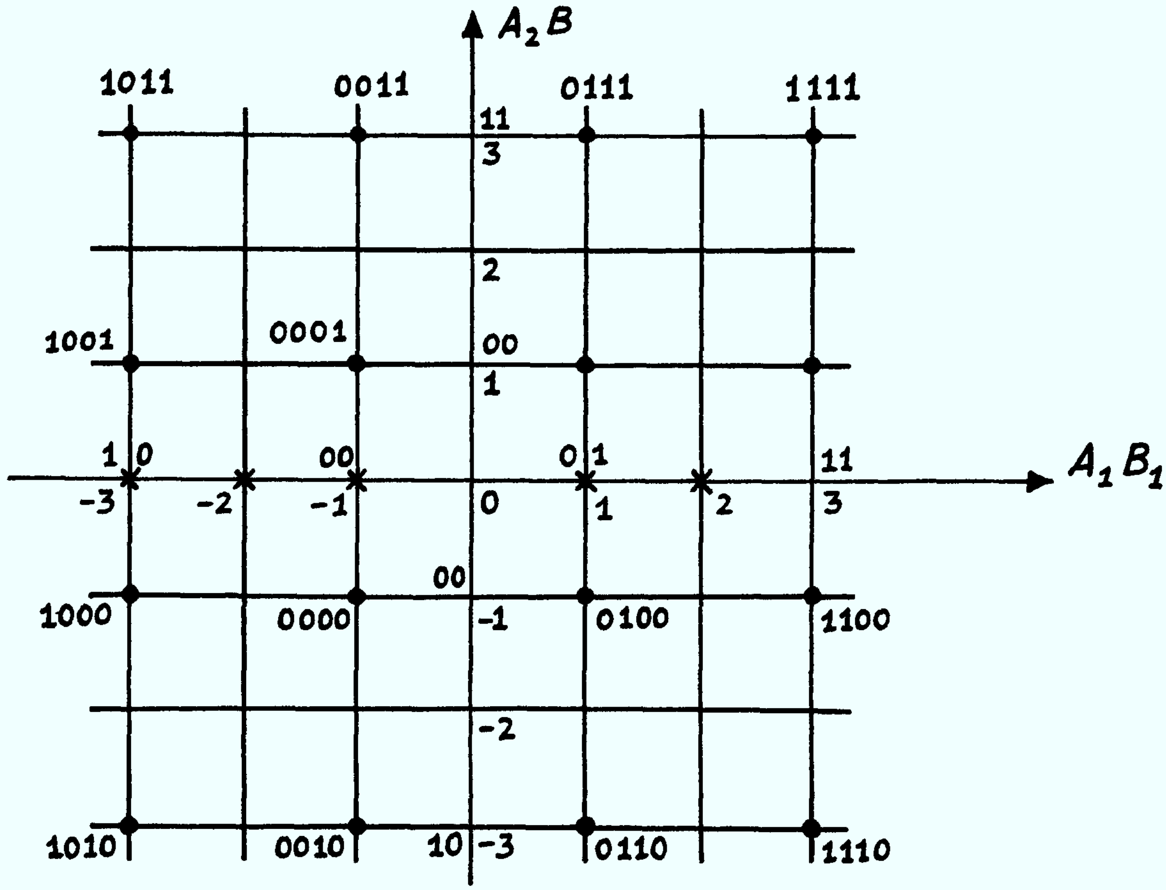


Figure A.1

Nous pouvons déduire de cette figure les conditions d'appartenance aux régions de décision associées à A_1 et B_1. Le signal reçu étant:

$$x(\tau) = a_o + jb_o - (b_o - ja_o)Q(\tau) + (X + jY) + (U + jV)$$

Alors:

Pour $A_1 = 0$ on a: $|R_e(x)| < 2$; pour $A_1 = 1$ on a $|R_e(x)| > 2$

pour $B_1 = 0$ on a: $R_e(x) < 0$; pour $B_1 = 1$ on a $R_e(x) > 0$

pour $A_2 = 0$ on a: $|I_m(x)| < 2$; pour $A_2 = 1$ on a $|I_m(x)| > 2$

pour $B_1 = 0$ on a: $I_m(x) < 0$; pour $B_1 = 1$ on a $I_m(x) > 0$

Étant donné que X et Y, $R_e(x)$ et $I_m(x)$ sont des variables aléatoires gaussiennes de variance σ^2 et de moyennes $\mu_x = a_o - b_o Q_o + X$ et $\mu_y = b_o + a_o Q_o + Y$. Pour chaque couple $A_i B_i$ ($i = 1, 2$), on doit évaluer la probabilité qu'il se situe hors de la zone de décision, ce qui revient à calculer des intégrales du type:

$$\frac{1}{\sigma\sqrt{2\pi}} \int_a^b \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)dx = \frac{1}{2}\left[\operatorname{erf}\left(\frac{b-\mu}{\sigma\sqrt{2}}\right) - \operatorname{erf}\left(\frac{a-\mu}{\sigma\sqrt{2}}\right)\right]$$

où

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_o^x \exp(-u^2)du$$

et

$$\operatorname{erfc}(x) = 1 - \operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty \exp(-u^2)du$$

Effectuons maintenant les calculs pour $A_i B_i$:

<u>**1° cas:**</u> $A_1 = B_1$ (soit $a_o = 3$)

$$\Pr[A_1 \text{ faux}] = \frac{1}{2}\left[\operatorname{erf}\left(\frac{2+\mu_x}{\sigma\sqrt{2}}\right) - \operatorname{erf}\left(\frac{-2+\mu_x}{\sigma\sqrt{x}}\right)\right]$$

$$= \frac{1}{2}\left[\operatorname{erf}\left(\frac{5-b_o Q_o + X}{\sigma\sqrt{2}}\right) - \operatorname{erf}\left(\frac{1-b_o Q_o + X}{\sigma\sqrt{2}}\right)\right]$$

$$Pr[B_1 \text{ faux}] = \frac{1}{2}\,[1 - \mathrm{erf}(\frac{\mu_x}{\sigma\sqrt{2}})] = \frac{1}{2}\,[1 - \mathrm{erf}(\frac{3-b_oQ_o+X}{\sigma\sqrt{2}})]$$

2° cas: $A_1 = 0,\ B_1 = 1$ (soit $a_o = 1$)

$$Pr[A_1 \text{ faux}] = 1 - \frac{1}{2}\,[\mathrm{erf}(\frac{2+\mu_x}{\sigma\sqrt{2}}) - \mathrm{erf}(\frac{-2+\mu_x}{\sigma\sqrt{2}})]$$

$$= 1 - \frac{1}{2}\,[\mathrm{erf}(\frac{3-b_oQ_o+X}{\sigma\sqrt{2}}) - \mathrm{erf}(\frac{-1-b_oQ_o+X}{\sigma\sqrt{2}})]$$

$$Pr[B_1 \text{ faux}] = \frac{1}{2}\,[1 - \mathrm{erf}(\frac{\mu_x}{\sigma\sqrt{2}})] = \frac{1}{2}\,[1 - \mathrm{erf}(\frac{1-b_oQ_o+X}{\sigma\sqrt{2}})]$$

3° cas: $A_1 = 0,\ B_1 = 0$ (soit $a_o = -1$)

$$Pr[A_1 \text{ faux}] = 1 - \frac{1}{2}\,[\mathrm{erf}(\frac{2+\mu_x}{\sigma\sqrt{2}}) - \mathrm{erf}(\frac{-2+\mu_x}{\sigma\sqrt{2}})]$$

$$= 1 - \frac{1}{2}\,[\mathrm{erf}(\frac{1-b_oQ_o+X}{\sigma\sqrt{2}}) - \mathrm{erf}(\frac{-3-b_oQ_o+X}{\sigma\sqrt{2}})]$$

4° cas: $A_1 = 1,\ B_1 = 0$ (soit $a_o = -3$)

$$Pr[A_1 \text{ faux}] = \frac{1}{2}\,[\mathrm{erf}(\frac{2+\mu_x}{\sigma\sqrt{2}}) - \mathrm{erf}(\frac{-2+\mu_x}{\sigma\sqrt{2}})]$$

$$= \frac{1}{2}\,[\mathrm{erf}(\frac{-1-b_oQ_o+X}{\sigma\sqrt{2}}) - \mathrm{erf}(\frac{-5-b_oQ_o+X}{\sigma\sqrt{2}})]$$

$$Pr[B_1 \text{ faux}] = \frac{1}{2}\,[1 + \mathrm{erf}(\frac{\mu_x}{\sigma\sqrt{2}})] = \frac{1}{2}\,[1 + \mathrm{erf}(\frac{-3-b_oQ_o+X}{\sigma\sqrt{2}})]$$

Avec $\alpha = -\,b_oQ_o + X$ et en utilisant la fonction erfc(x), les probabilités conditionnelles de A_1 et B_1 par rapport à X et b_o sont alors:

$$\Pr\{A_1 \text{faux} | X_1, b_o\} = \frac{1}{4}\ \{\text{erfc}(\frac{1+\alpha}{\sigma\sqrt{2}}) + \text{erfc}(\frac{1-\alpha}{\sigma\sqrt{x}}) + \frac{1}{2}\ \text{erfc}(\frac{3+\alpha}{\sigma\sqrt{2}})$$

$$- \frac{1}{2}\ [\text{erfc}(\frac{3-\alpha}{\sigma\sqrt{2}}) + \text{erfc}(\frac{5+\alpha}{\sigma\sqrt{2}}) + \text{erfc}(\frac{5-\alpha}{\sigma\sqrt{2}})]\}$$

$$\Pr\{B_1 \text{faux} | X_1, b_o\} = \frac{1}{8}\ [\text{erfc}(\frac{1+\alpha}{\sigma\sqrt{2}}) + \text{erfc}(\frac{1-\alpha}{\sigma\sqrt{x}}) + \text{erfc}(\frac{3+\alpha}{\sigma\sqrt{2}}) + \text{erfc}(\frac{3-\alpha}{\sigma\sqrt{2}})]$$

Pour obtenir $P_e^{A_1} = P[A_1 \text{Faux}]$ et $P_e^{B_1} = P\ [B_1 \text{faux}]$, il suffit de calculer les expressions:

$$P_e^{A_1} = \frac{1}{4} \sum_{b_o} \int_{-\infty}^{\infty} P_r(A_1 \text{faux} | X_1, b_o) dF(x)$$

$$P_e^{B_1} = \frac{1}{4} \sum_{b_o} \int_{-\infty}^{\infty} P_r(B_1 \text{faux} | X_1, b_o) dF(x)$$

Les calculs pour $P_e^{A_2}$ et $P_e^{B_2}$ se réalisent de la même manière en remplaçant X par Y, a_o par b_o et $b_o Q_o$ par $a_o Q_o$. Quand le rapport signal à bruit est élevé, typiquement de 20 dB ($\text{soit}\ A/\sigma\sqrt{2} \sim 3$) et que nous sommes dans le pire des cas pour l'interférence intersymbole (soit $\alpha = -1$), il vient:

$$\text{erfc}(\frac{5+\alpha}{\sigma\sqrt{2}}) \sim 1.35 \times 10^{-64} \quad \text{et} \quad \text{erfc}(\frac{3+\alpha}{\sigma\sqrt{2}}) \sim 2.15 \times 10^{-17}$$

Ces termes sont négligeables devant $\text{erfc}(\frac{1+\alpha}{\sigma\sqrt{2}}) \sim 0.5$. L'équation (1) devient:

$$P_e^{A_1} = \frac{1}{4} \sum_{b_o} \frac{1}{2} \int_{-\infty}^{\infty} \text{erfc}(\frac{1+\alpha}{\sigma\sqrt{2}}) dF(x)$$

$$P_e^{B_1} = \frac{1}{4} \sum_{b_o} \frac{1}{4} \int_{-\infty}^{\infty} \text{erfc}(\frac{1+\alpha}{\sigma\sqrt{2}}) dF(x)$$

d'où

$$P_e^{A_1} = 2\ P_e^{B_1}.$$

ANNEXE 2

RÉSULTATS NUMÉRIQUES
(hypothèse quasi-stationnaire)

Définition des paramètres:

SIR : rapport signal à interférence en dB (ou SJR)

SNR : rapport signal à bruit

Y : déviation de fréquence RMS normalisée à B/2

X (position FM): écart de fréquence entre les porteuses FM et QAM normalisé à B/2

DSNR : augmentation du SNR nécessaire pour conserver une valeur constante de P_e.

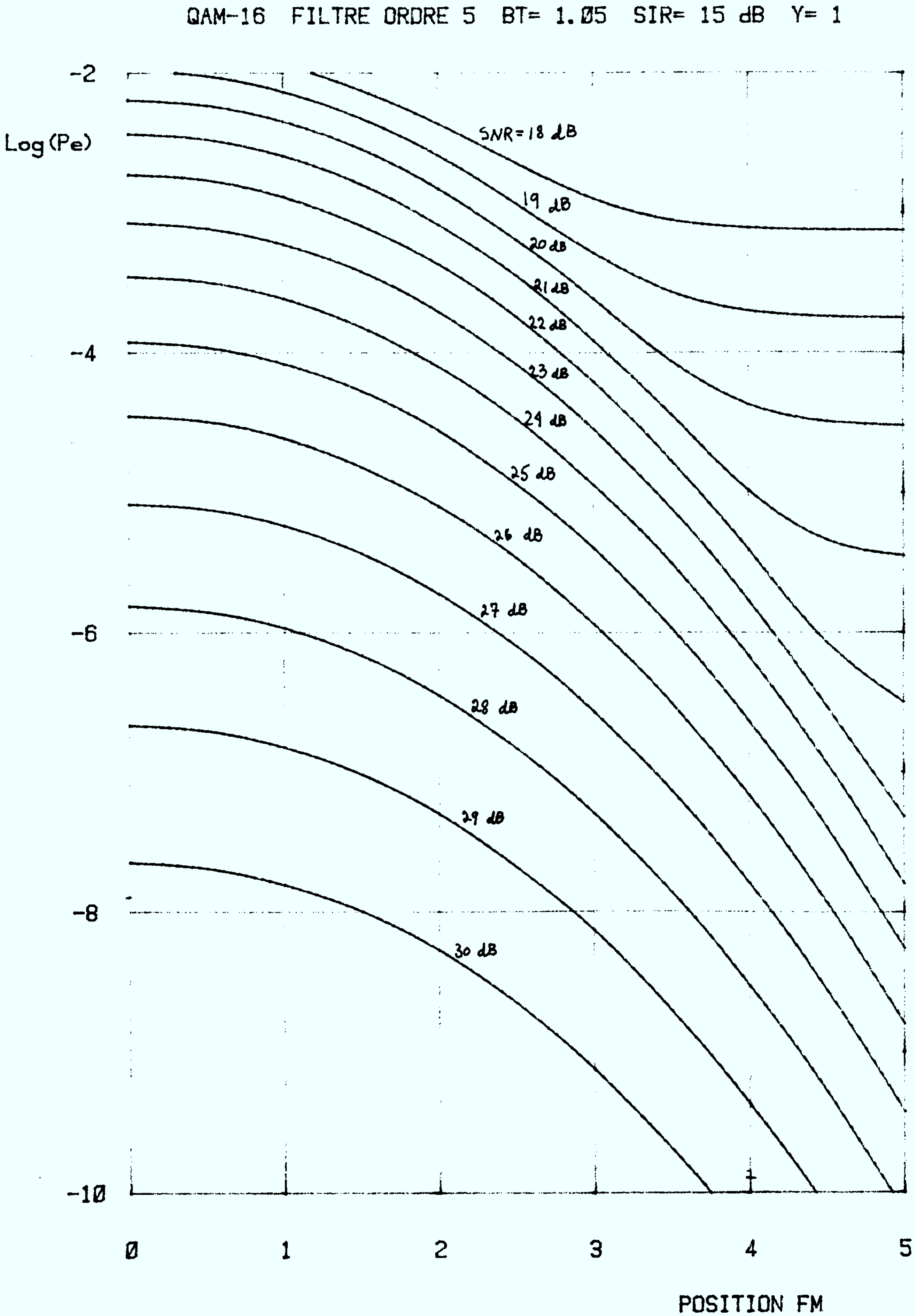

QAM-16 FILTRE ORDRE 5 BT= 1.05 SIR= 15 dB Y= 1
Log(Pe)
-2
-4
-6
-8
-10
SNR=18 dB
19 dB
20 dB
21 dB
22 dB
23 dB
24 dB
25 dB
26 dB
27 dB
28 dB
29 dB
30 dB
0
1
2
3
4
5
POSITION FM

FIGURE A.2.1

QAM-16 FILTRE ORDRE 5 BT= 1.05 SIR= 15 dB Y= 2

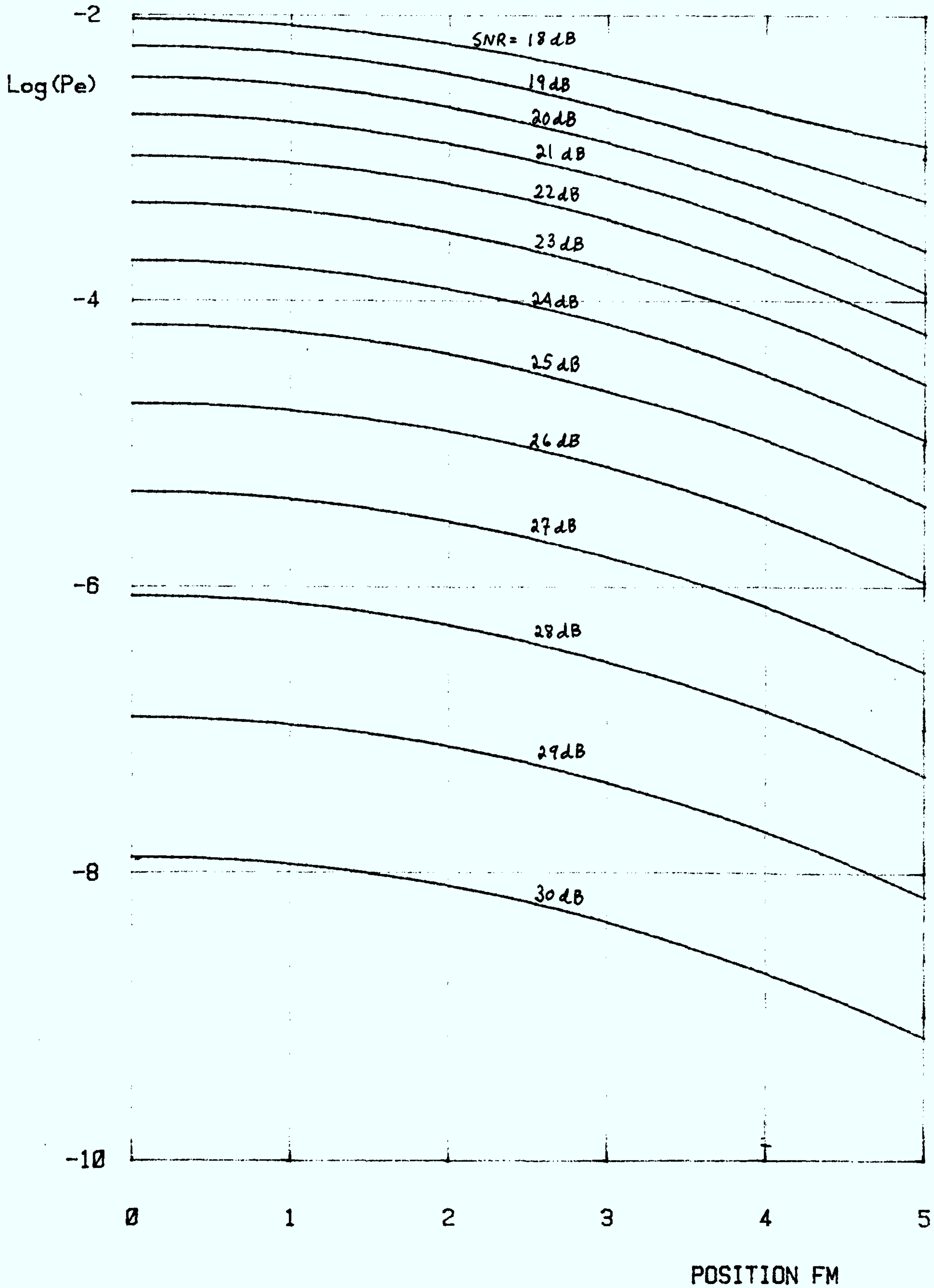


FIGURE A.2.2

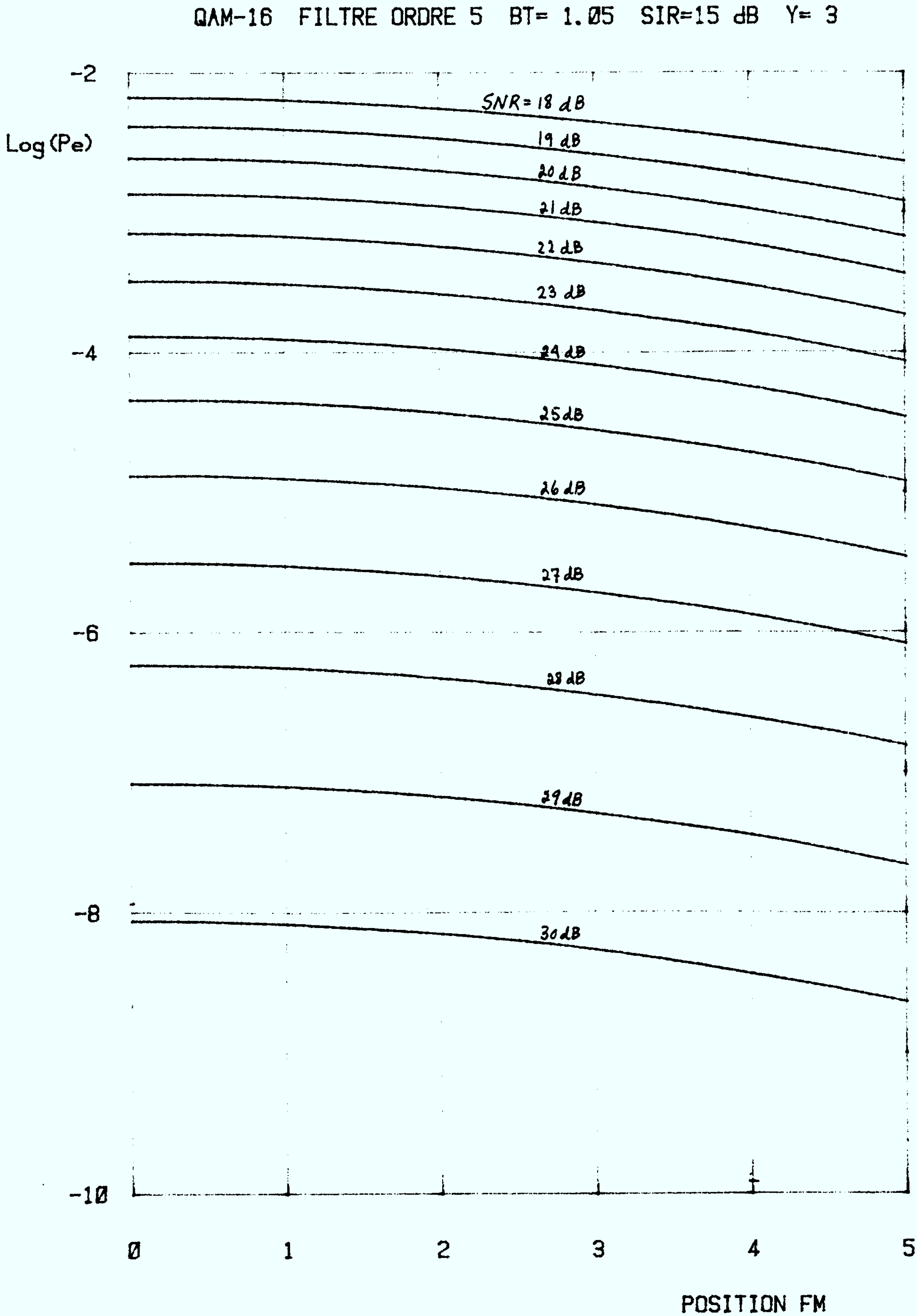

QAM-16 FILTRE ORDRE 5 BT= 1.05 SIR=15 dB Y= 3
-2
-4
-6
-8
-10
Log (Pe)
SNR = 18 dB
19 dB
20 dB
21 dB
22 dB
23 dB
24 dB
25 dB
26 dB
27 dB
28 dB
29 dB
30 dB
0
1
2
3
4
5
POSITION FM

FIGURE A.2.3

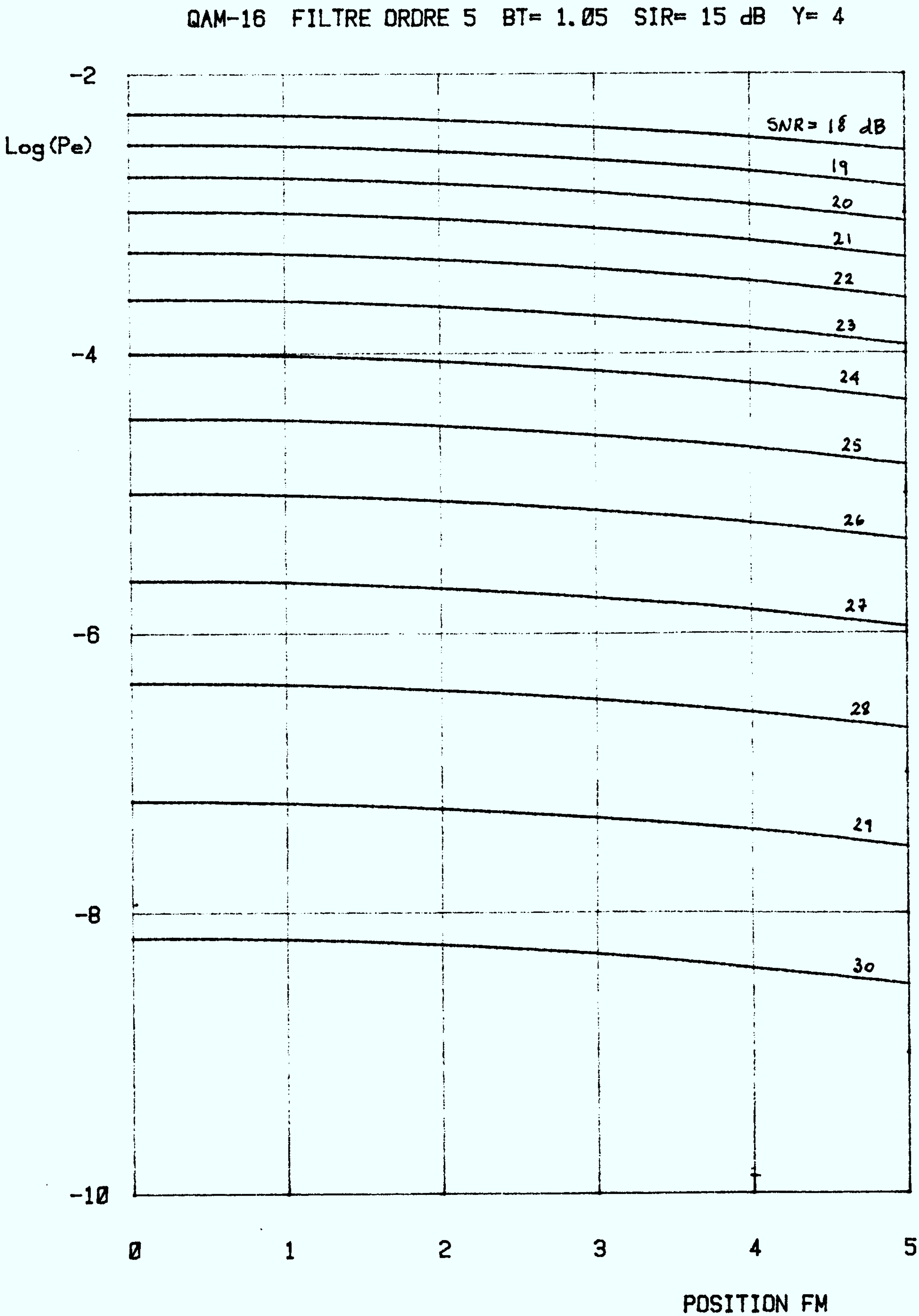

QAM-16 FILTRE ORDRE 5 BT= 1.05 SIR= 15 dB Y= 4
Log(Pe)
-2
-4
-6
-8
-10
SNR= 18 dB
19
20
21
22
23
24
25
26
27
28
29
30
0
1
2
3
4
5
POSITION FM

FIGURE A.2.4

QAM-16 FILTRE ORDRE 5 BT= 1.05 SIR= 15 dB Y= 5

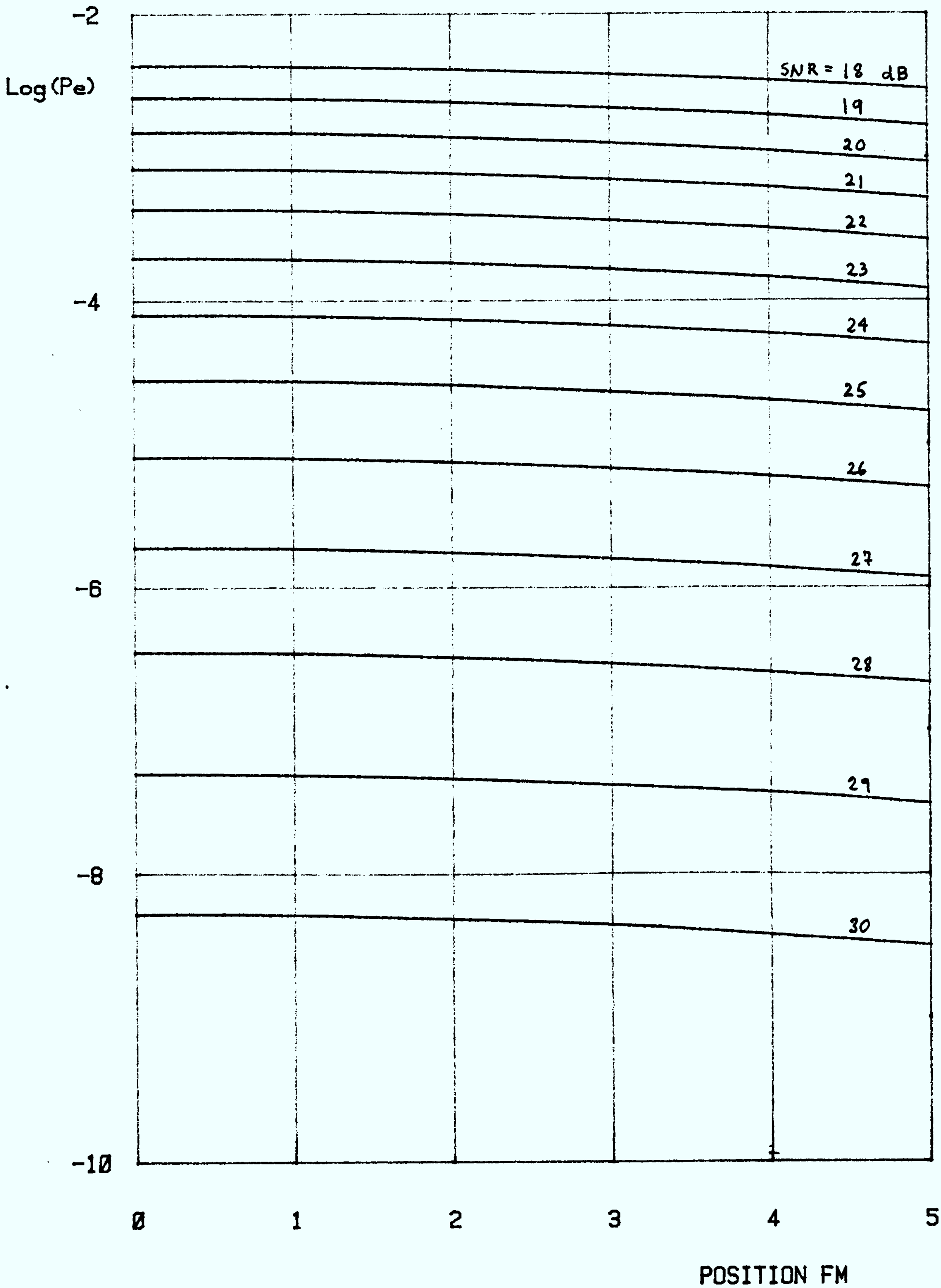


FIGURE A.2.5

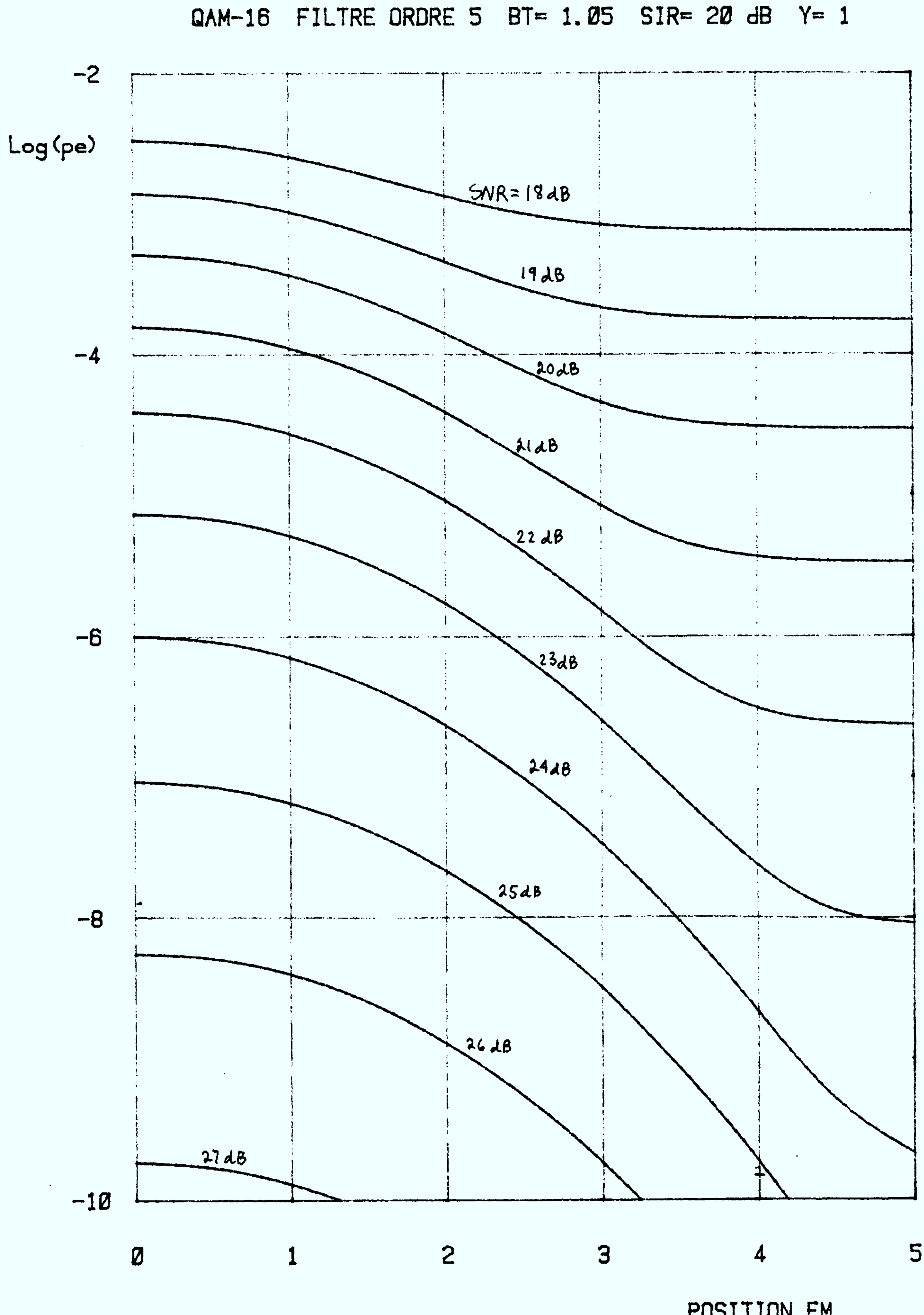

QAM-16 FILTRE ORDRE 5 BT= 1.05 SIR= 20 dB Y= 1
Log(pe)
-2
-4
-6
-8
-10
SNR=18dB
19dB
20dB
21dB
22dB
23dB
24dB
25dB
26dB
27dB
0
1
2
3
4
5
POSITION FM

FIGURE A.2.6

QAM-16 FILTRE ORDRE 5 BT= 1.Ø5 SIR= 25 dB Y= 1

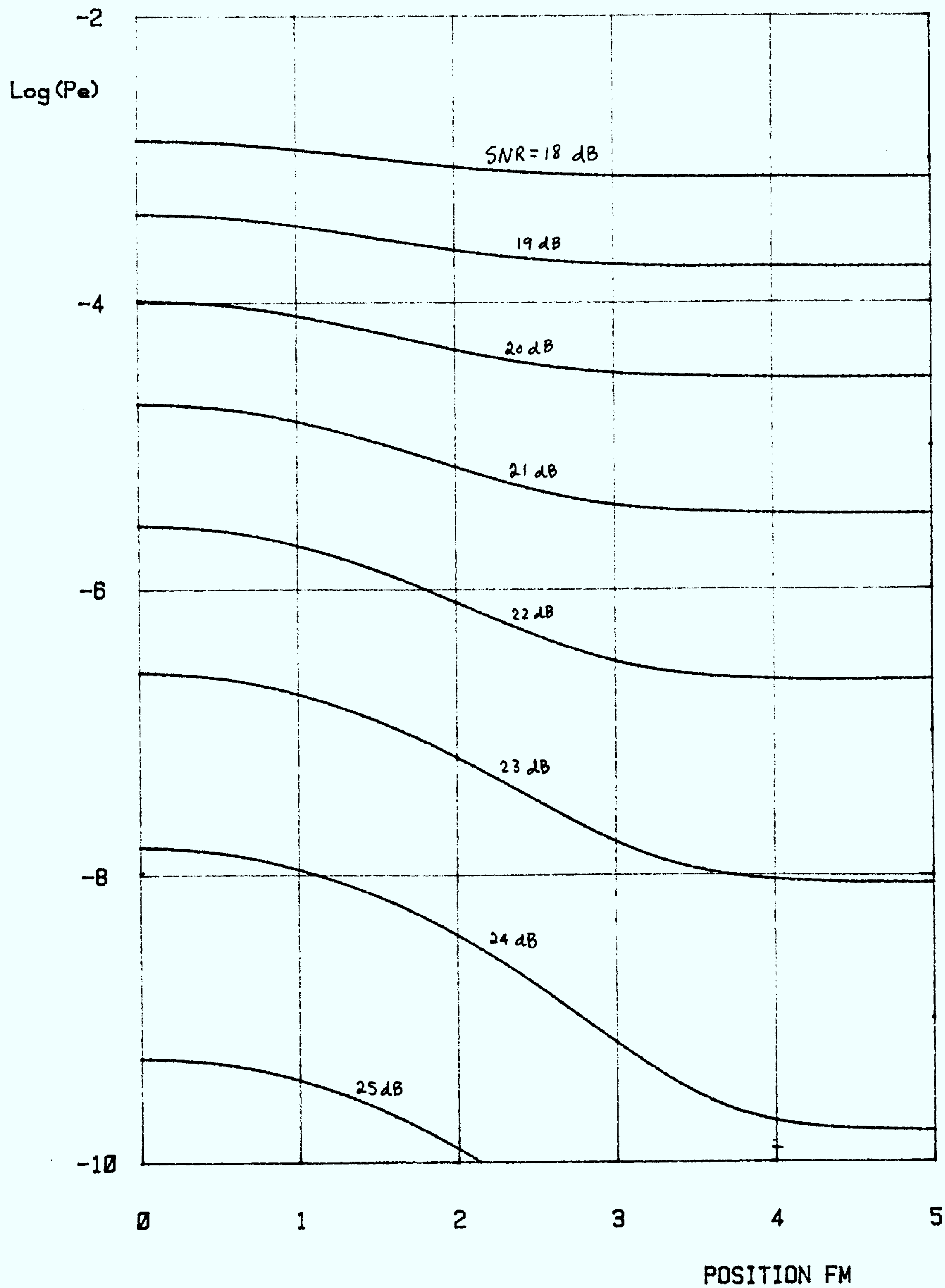


FIGURE A.2.7

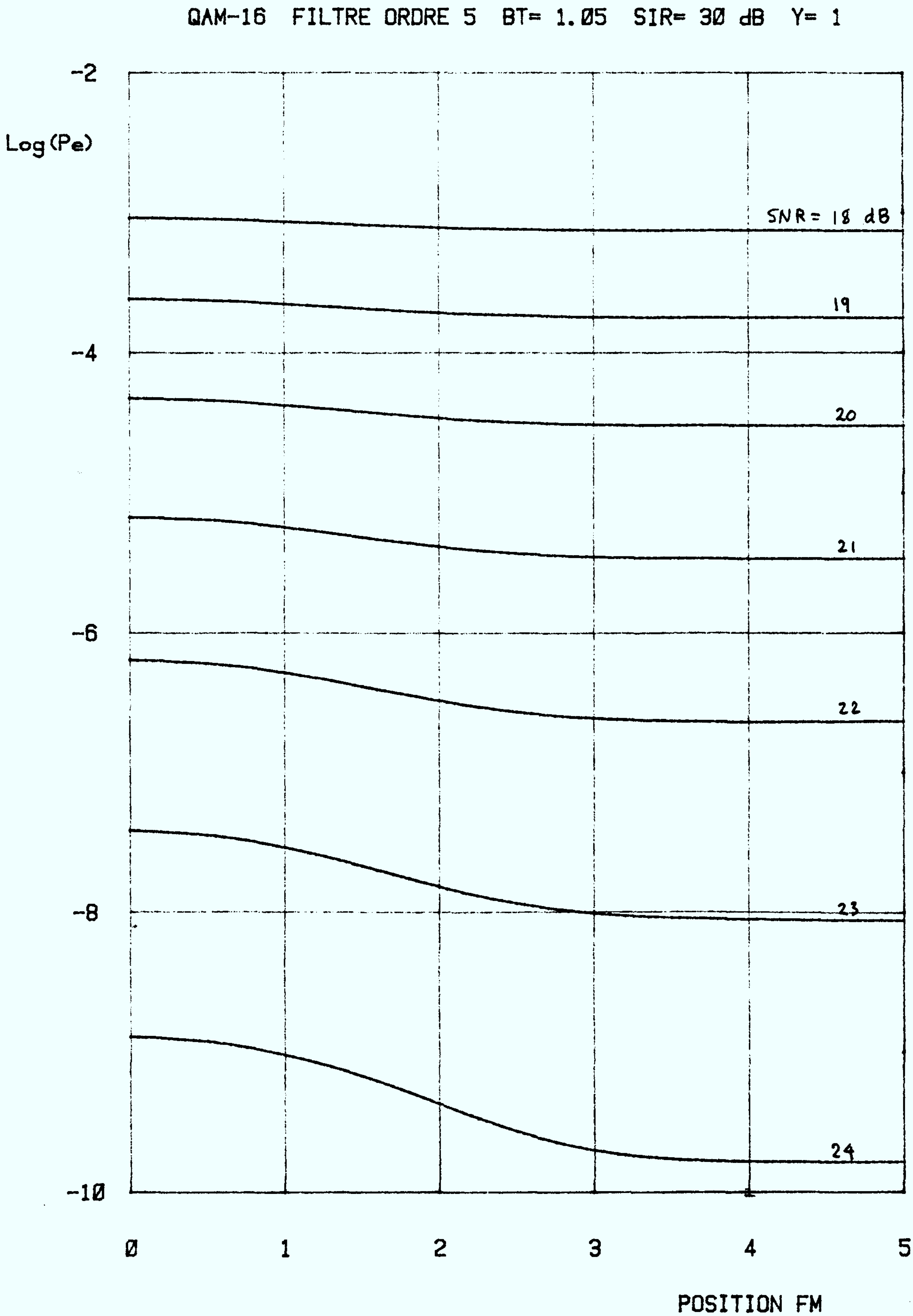

QAM-16 FILTRE ORDRE 5 BT= 1.05 SIR= 30 dB Y= 1
-2
Log(Pe)
SNR= 18 dB
19
-4
20
21
-6
22
23
-8
24
-10
0
1
2
3
4
5
POSITION FM

FIGURE A.2.8

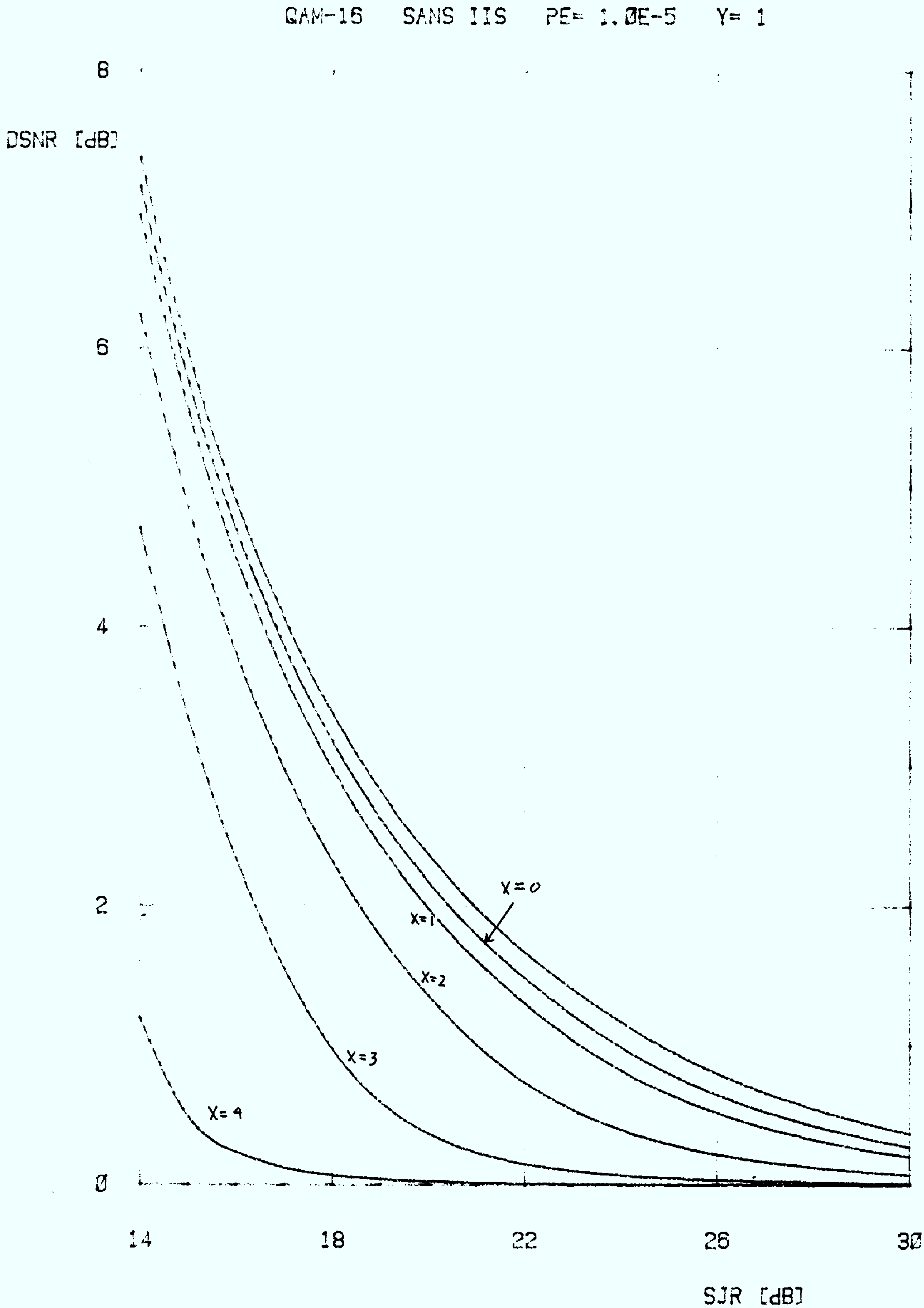

QAM-16 SANS IIS PE= 1.ØE-5 Y= 1
DSNR [dB]
8
6
4
2
Ø
14
18
22
26
3Ø
SJR [dB]
X=0
X=1
X=2
X=3
X=4

FIGURE A.2.9

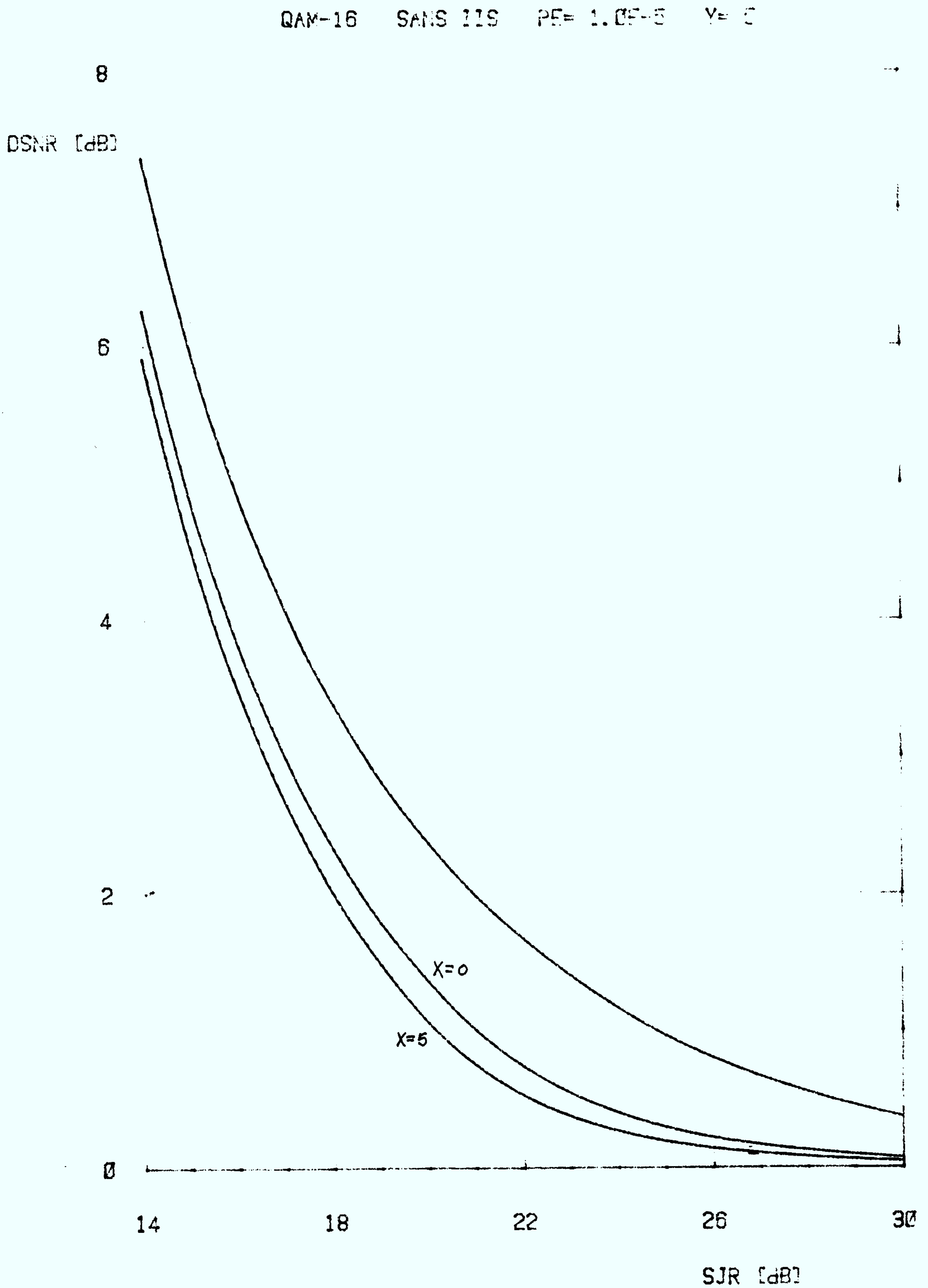

QAM-16 SANS IIS PE= 1.0E-5 Y= 0
8
DSNR [dB]
6
4
2
0
X=0
X=5
14
18
22
26
30
SJR [dB]

FIGURE A.2.10

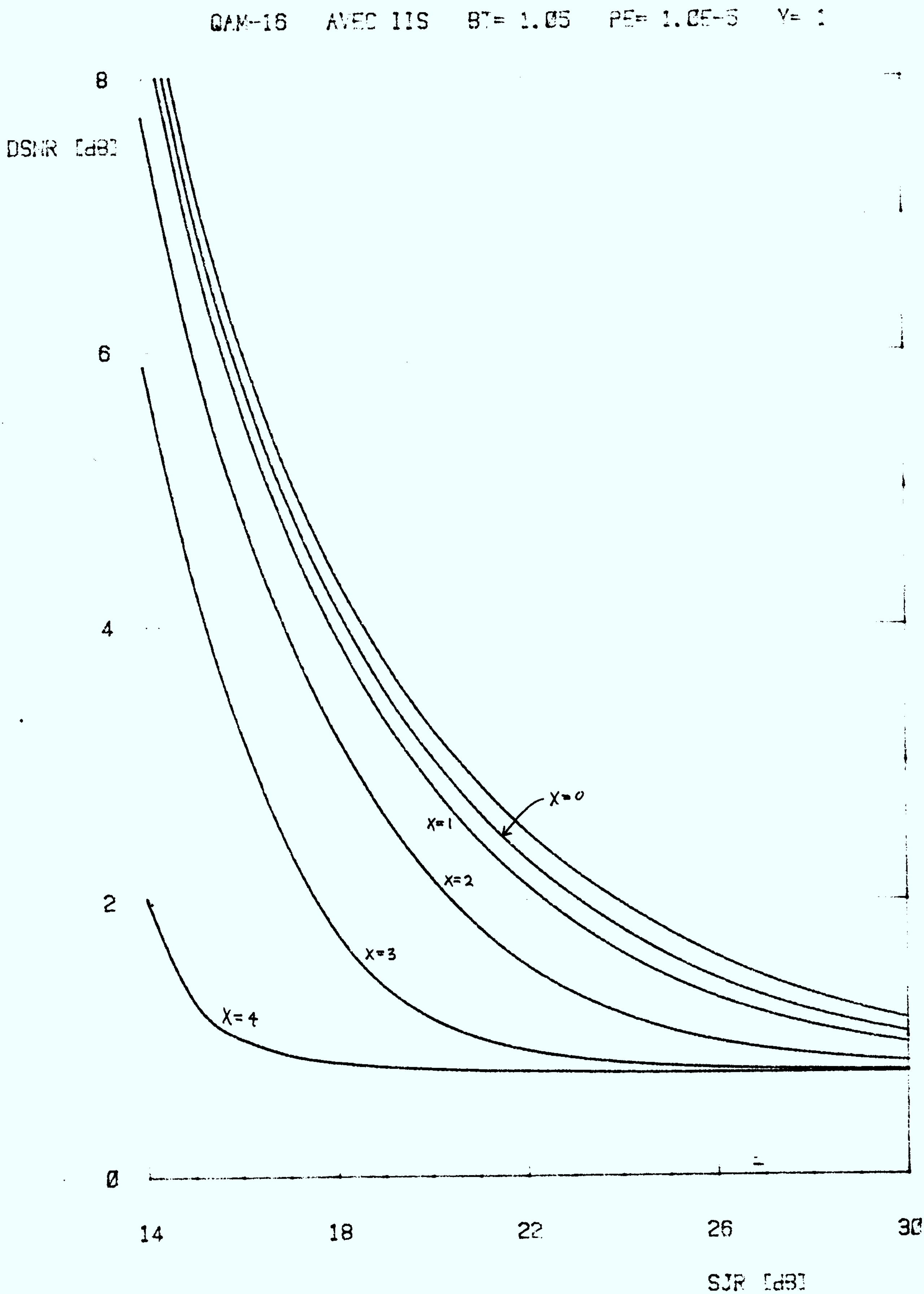

QAM-16 AVEC IIS BT= 1.Ø5 PE= 1.ØE-5 Y= 1
8
DSNR [dB]
6
4
2
Ø
14
18
22
26
30
SJR [dB]
X=0
X=1
X=2
X=3
X=4

FIGURE A.2.11

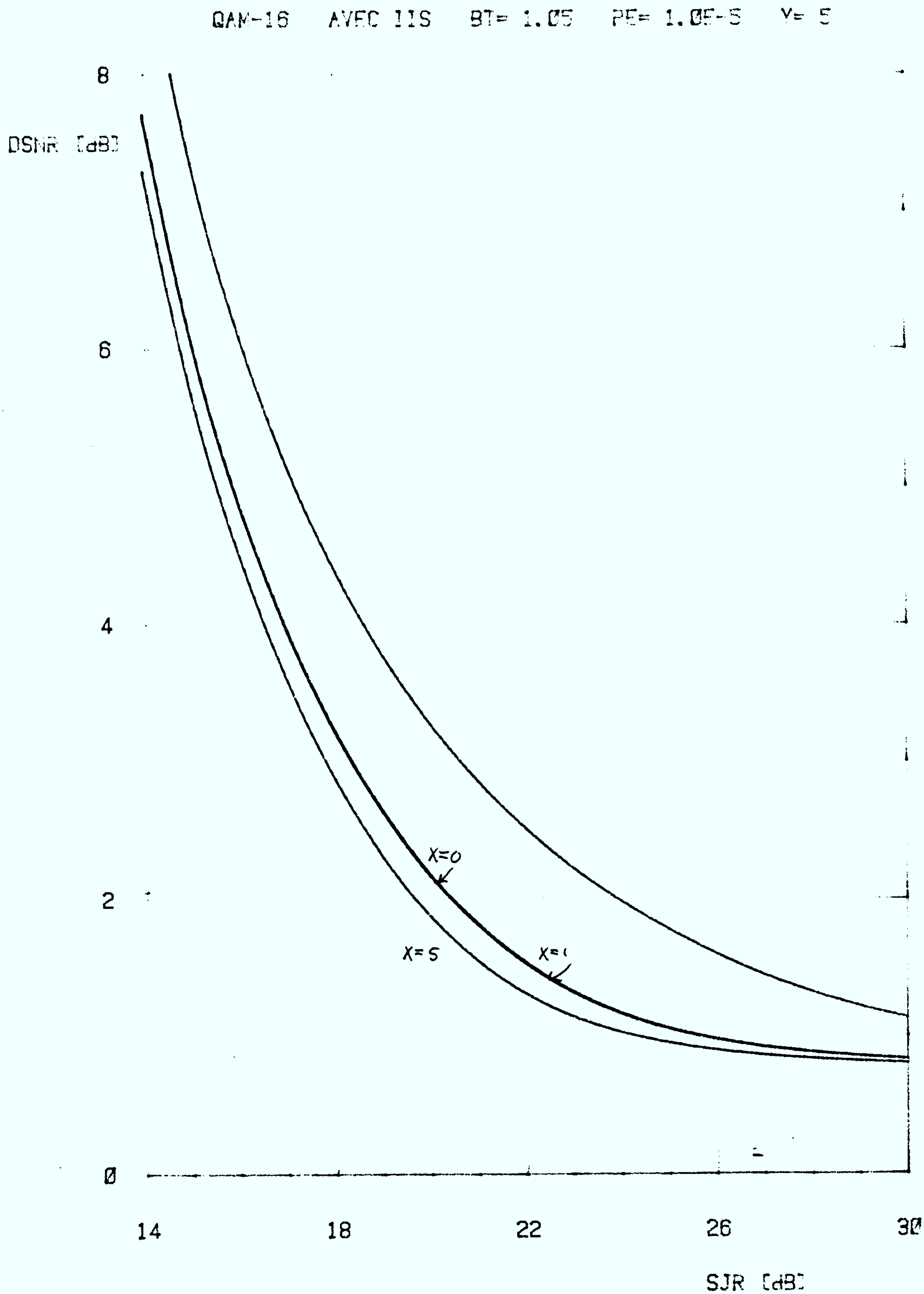

QAM-16 AVEC IIS BT= 1.05 PE= 1.0E-5 Y= 5
8
DSNR [dB]
6
4
2
0
X=0
X=5
14
18
22
26
30
SJR [dB]

FIGURE A.2.12

ANNEXE 3

RÉSULTATS NUMÉRIQUES
(hypothèse gaussienne)

Définition des paramètres:

SNR : rapport signal à bruit en dB

SIR : rapport signal à interférence en dB

DF : écart-type de la densité spectrale FM normalisé à B/2

FC : écart de fréquence entre les porteuses FM et QAM normalisé à B/2

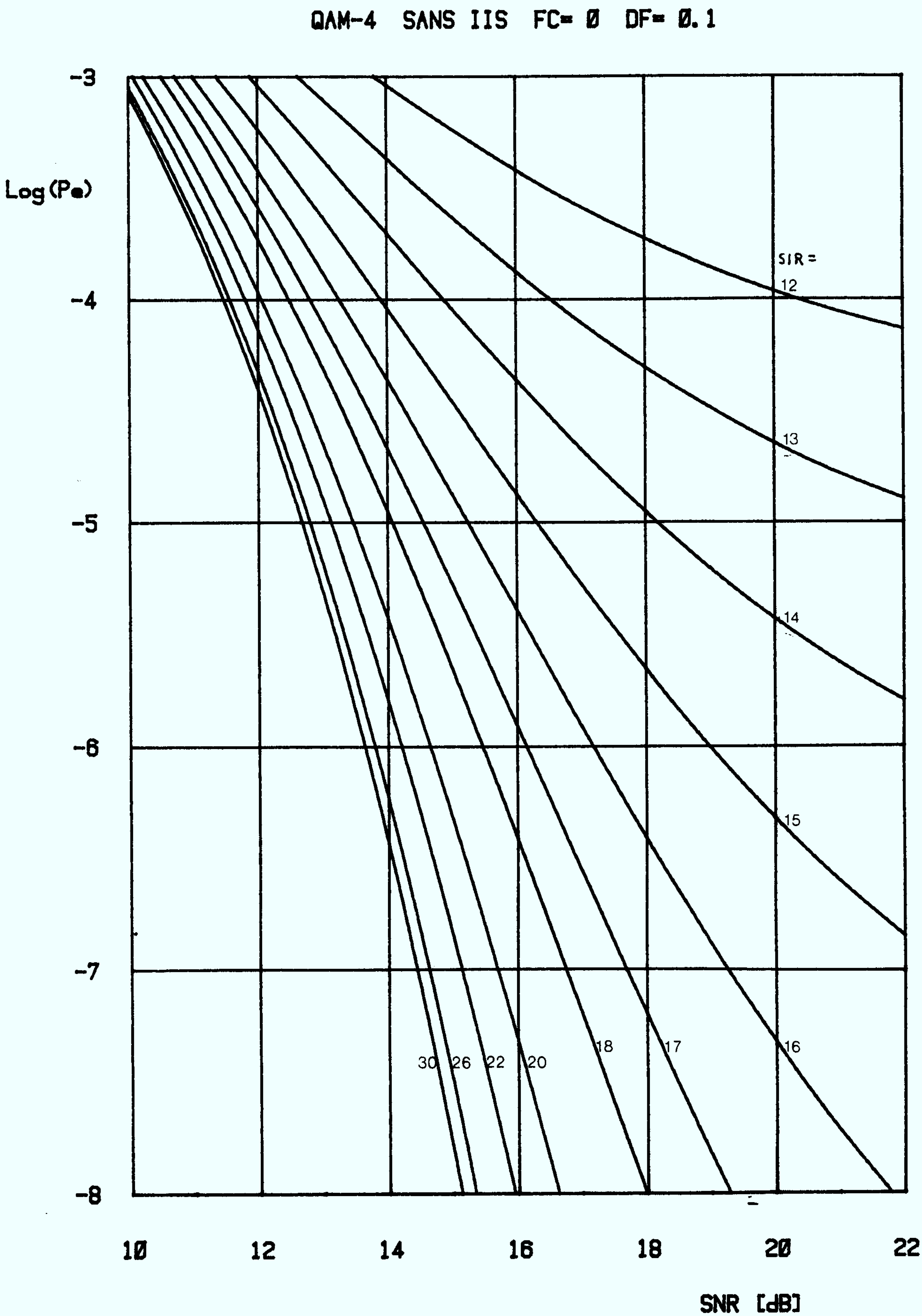

QAM-4 SANS IIS FC= Ø DF= Ø.1
Log(Pe)
-3
-4
-5
-6
-7
-8
SIR =
12
13
14
15
16
17
18
20
22
26
30
10
12
14
16
18
20
22
SNR [dB]

FIGURE A.3.1

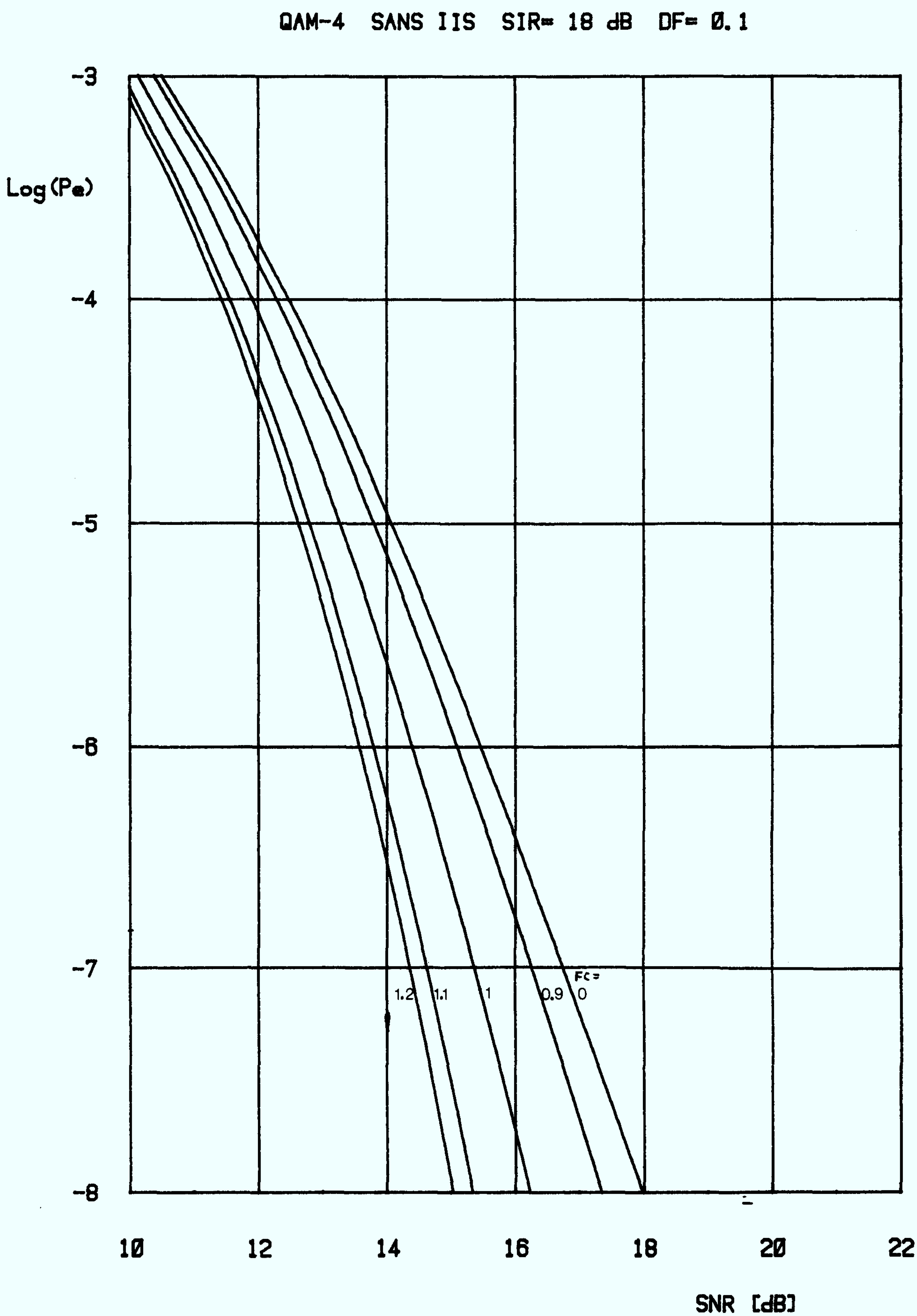

QAM-4 SANS IIS SIR= 18 dB DF= 0.1
Log(Pe)
-3
-4
-5
-6
-7
-8
10
12
14
16
18
20
22
SNR [dB]
FC=
1.2
1.1
1
0.9
0

FIGURE A.3.2

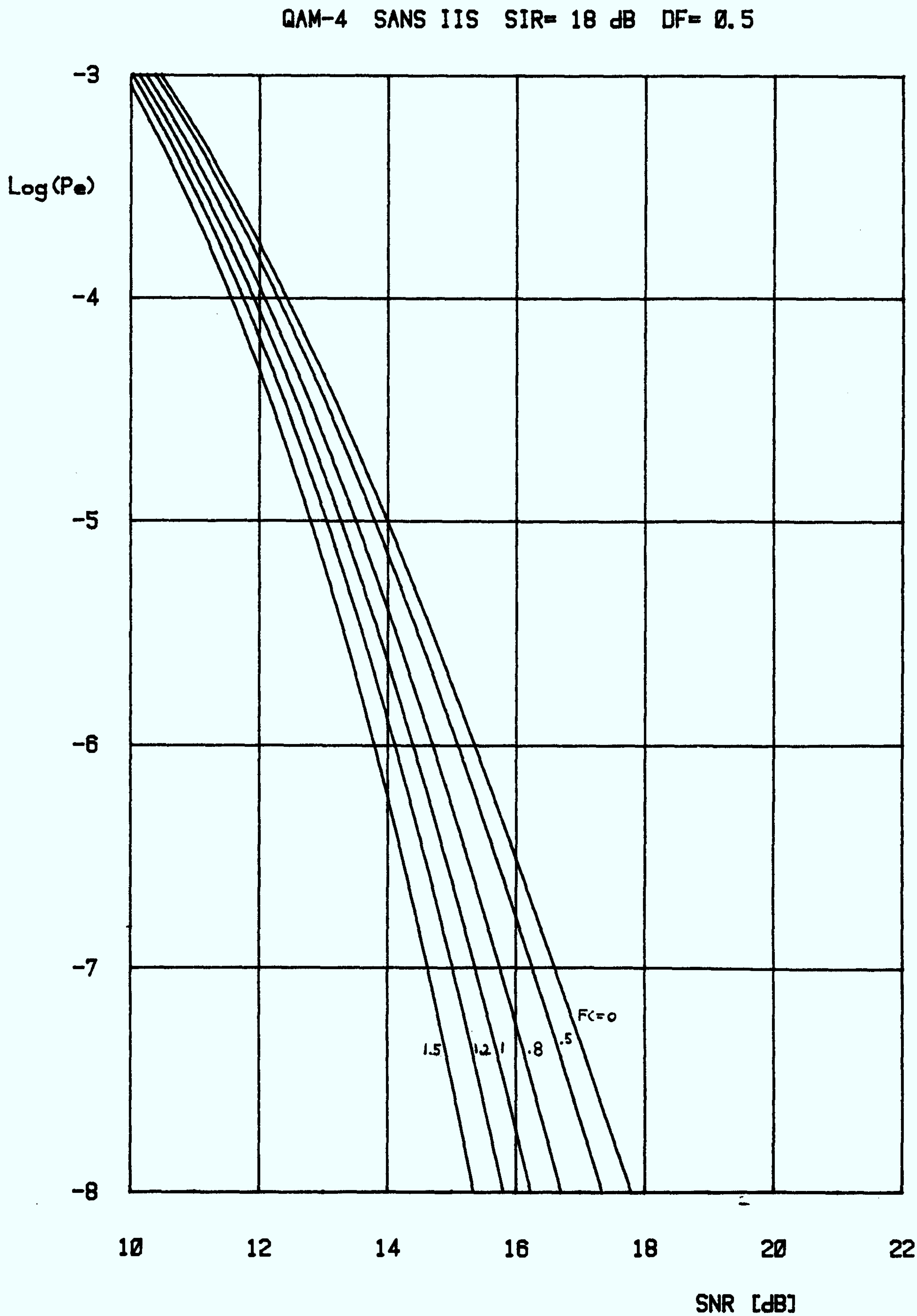

QAM-4 SANS IIS SIR= 18 dB DF= 0.5
-3
-4
-5
-6
-7
-8
Log(Pe)
10
12
14
16
18
20
22
SNR [dB]
FC=0
.5
.8
1
1.2
1.5

FIGURE A.3.3

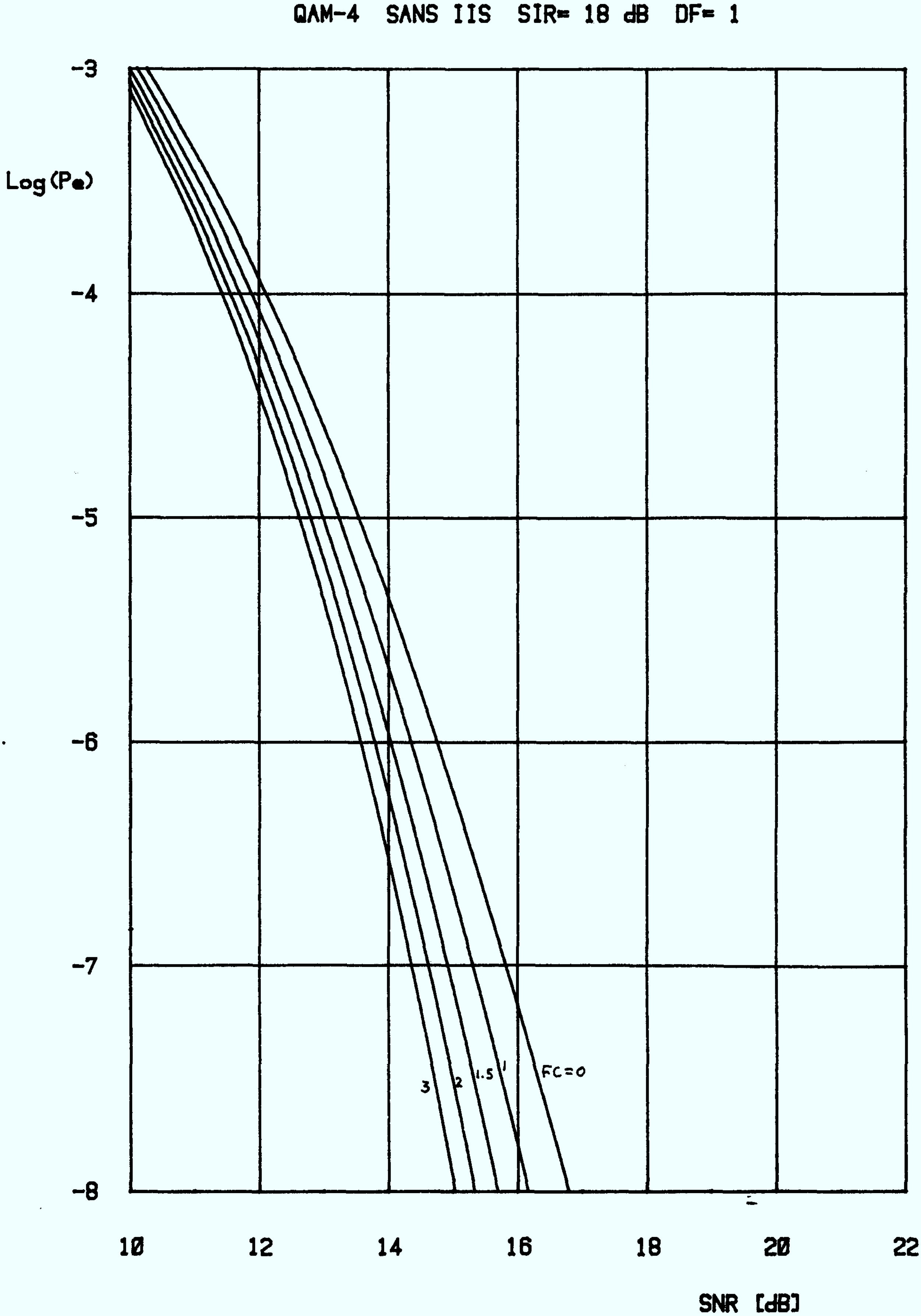

QAM-4 SANS IIS SIR= 18 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
3
2
1.5
1
FC=0
10
12
14
16
18
20
22
SNR [dB]

FIGURE A.3.4

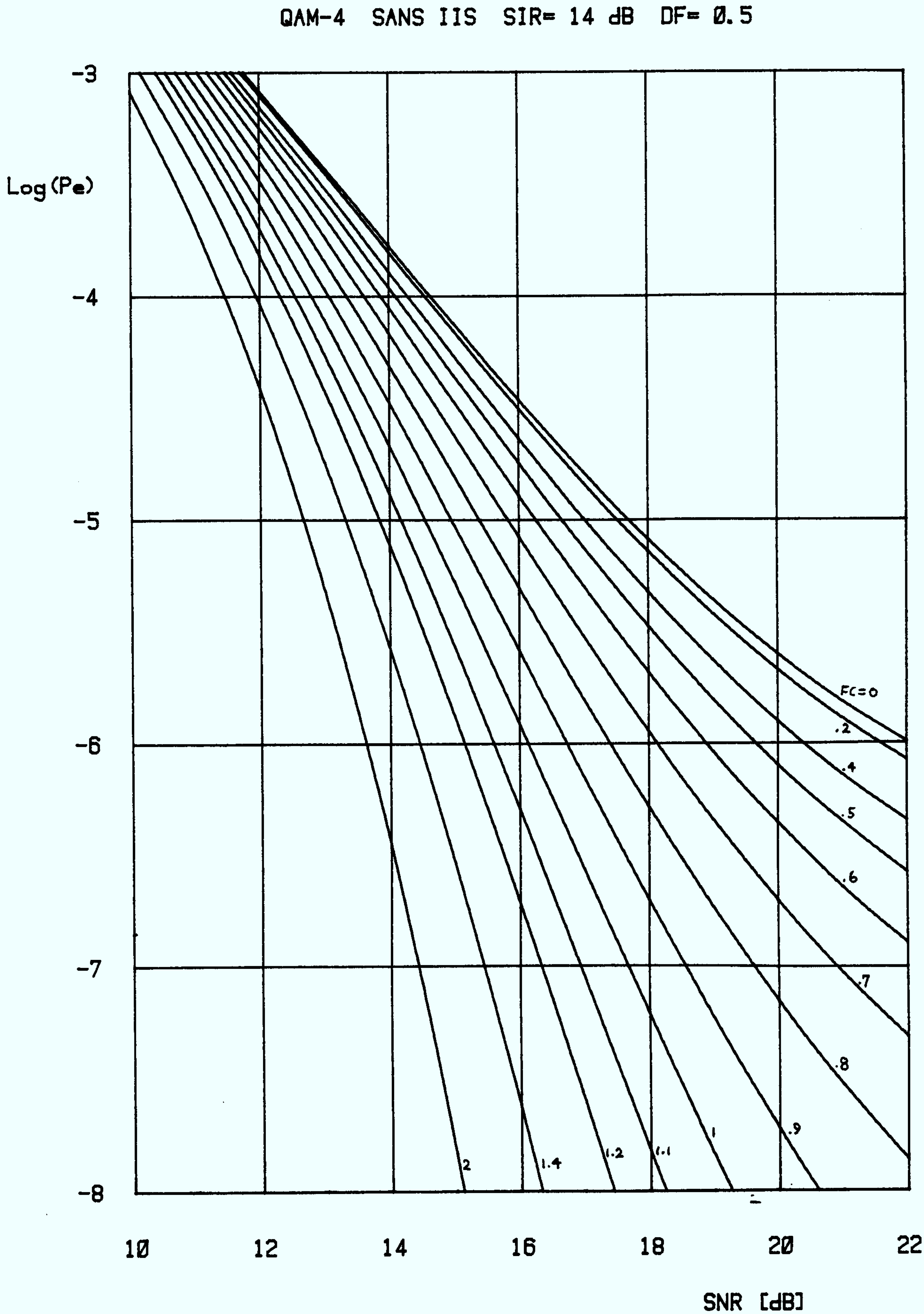

QAM-4 SANS IIS SIR= 14 dB DF= 0.5
Log (Pe)
-3
-4
-5
-6
-7
-8
10
12
14
16
18
20
22
SNR [dB]
FC=0
.2
.4
.5
.6
.7
.8
.9
1
1.1
1.2
1.4
2

FIGURE A.3.5

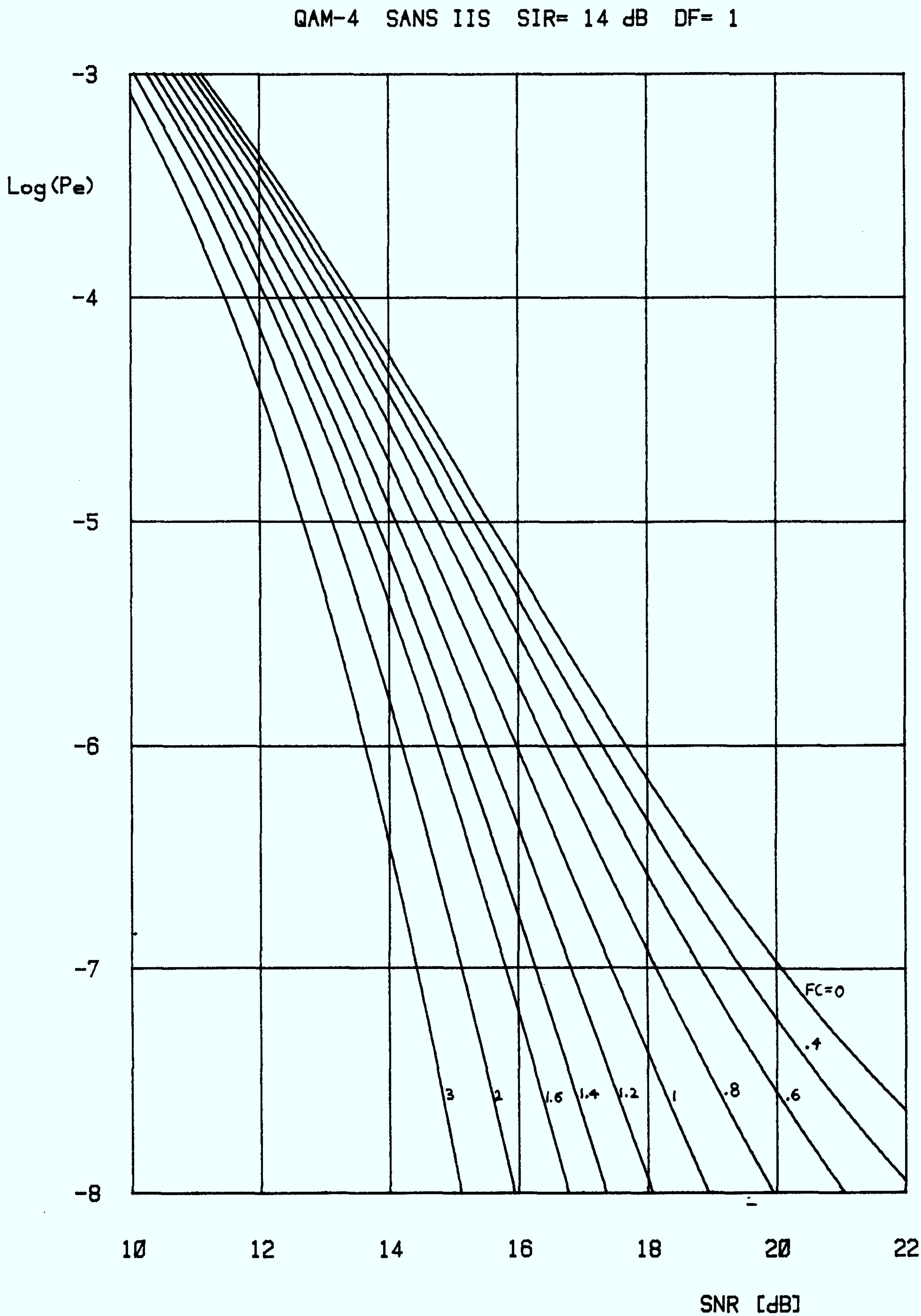

QAM-4 SANS IIS SIR= 14 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
10
12
14
16
18
20
22
SNR [dB]
FC=0
.4
.6
.8
1
1.2
1.4
1.6
2
3

FIGURE A.3.6

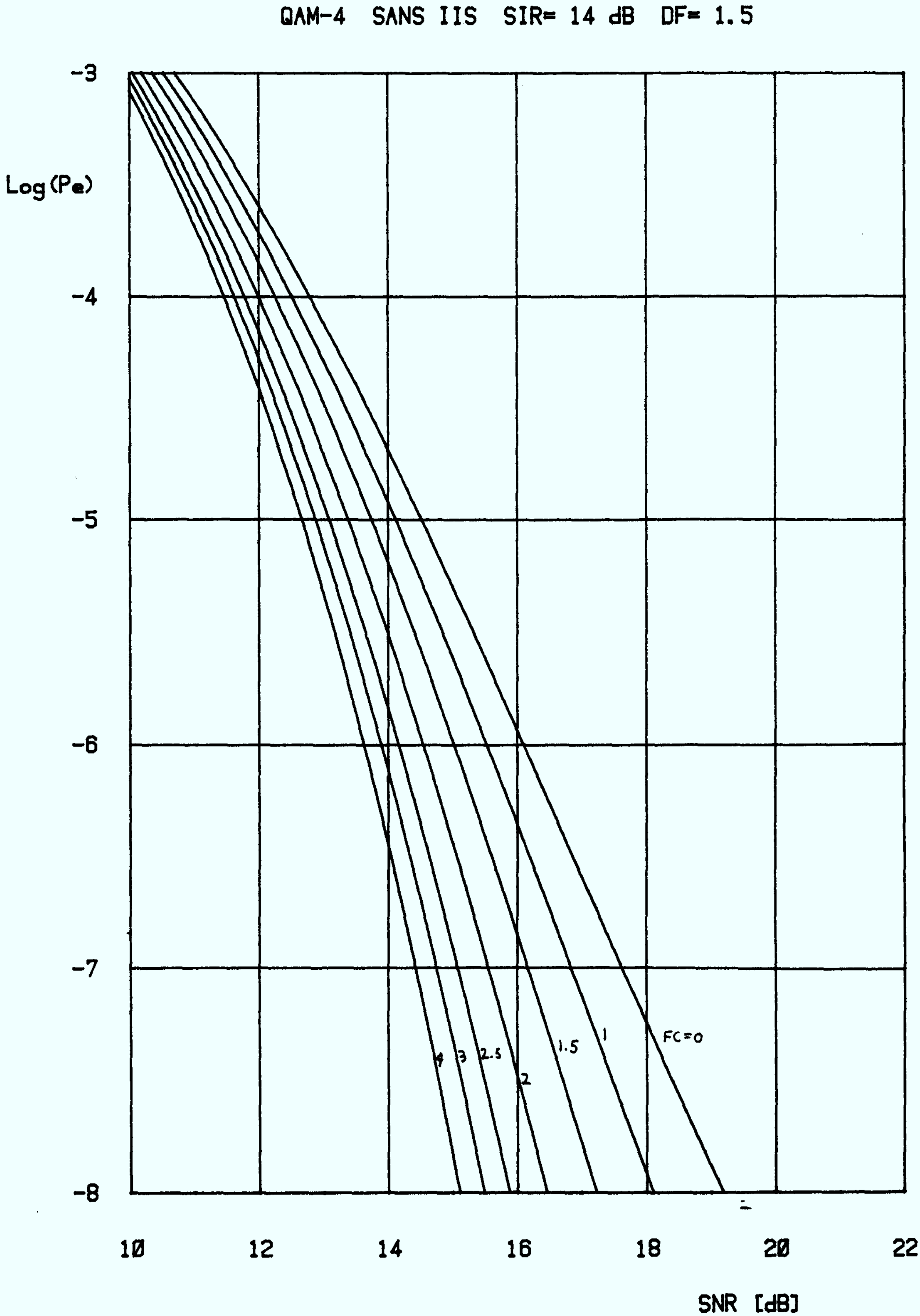

QAM-4 SANS IIS SIR= 14 dB DF= 1.5
Log (Pe)
-3
-4
-5
-6
-7
-8
10
12
14
16
18
20
22
SNR [dB]
4
3
2.5
2
1.5
1
FC=0

FIGURE A.3.7

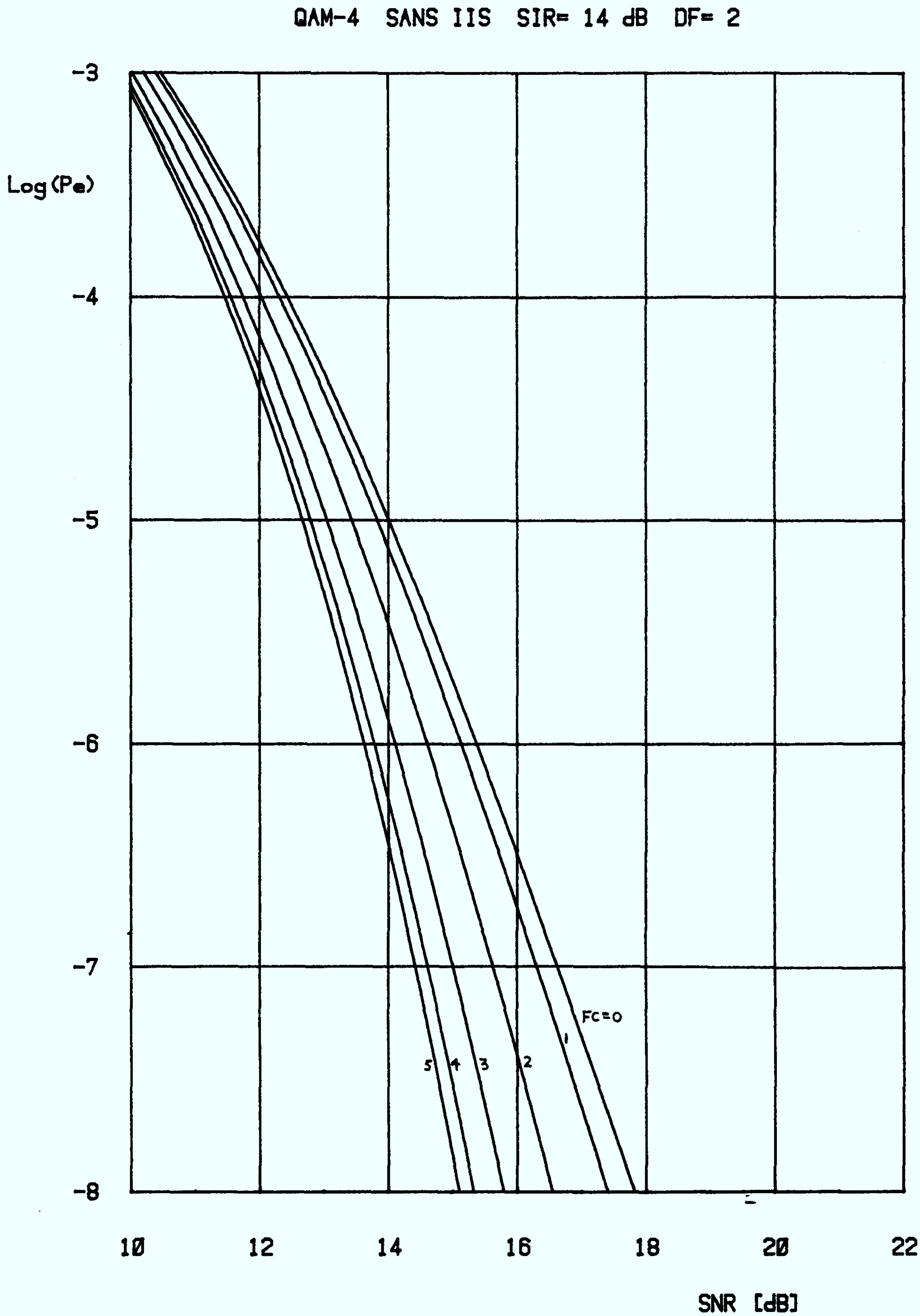

QAM-4 SANS IIS SIR= 14 dB DF= 2
Log(Pe)
-3
-4
-5
-6
-7
-8
10
12
14
16
18
20
22
SNR [dB]
Fc=0
1
2
3
4
5

FIGURE A.3.8

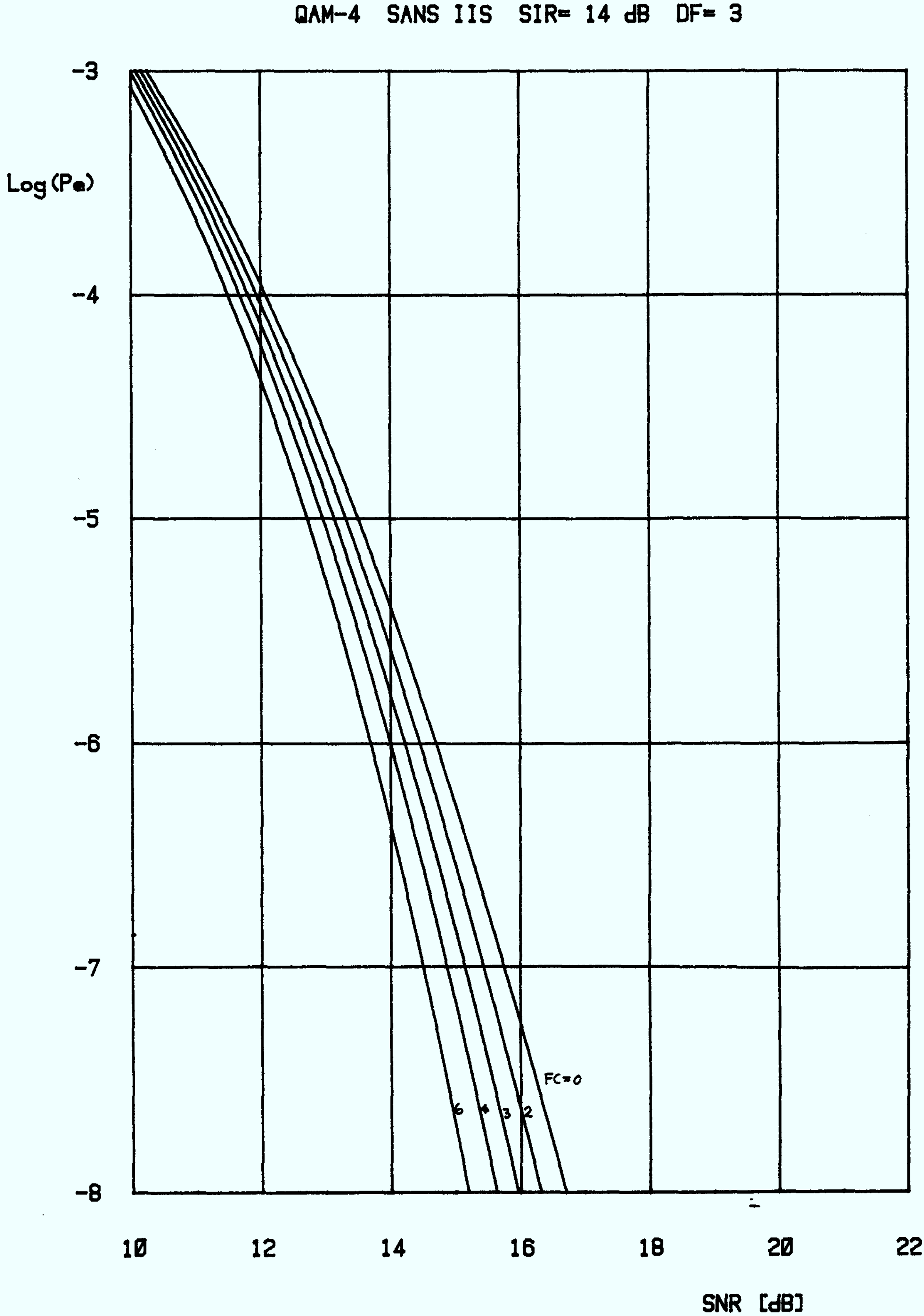

QAM-4 SANS IIS SIR= 14 dB DF= 3
Log(Pe)
-3
-4
-5
-6
-7
-8
FC=0
6
4
3
2
10
12
14
16
18
20
22
SNR [dB]

FIGURE A.3.9

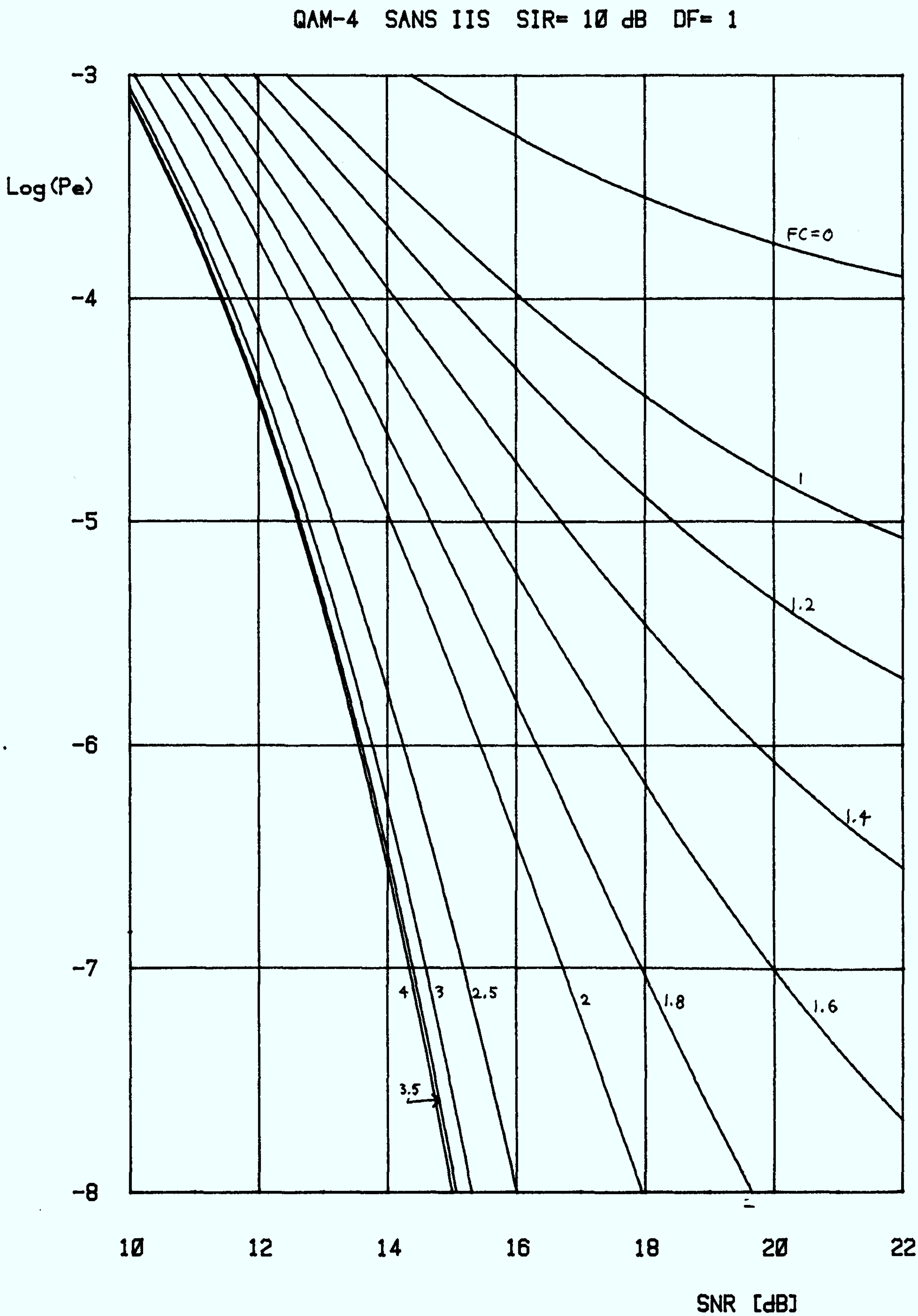

QAM-4 SANS IIS SIR= 10 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
FC=0
1
1.2
1.4
1.6
1.8
2
2.5
3
3.5
4
10
12
14
16
18
20
22
SNR [dB]

FIGURE A.3.10

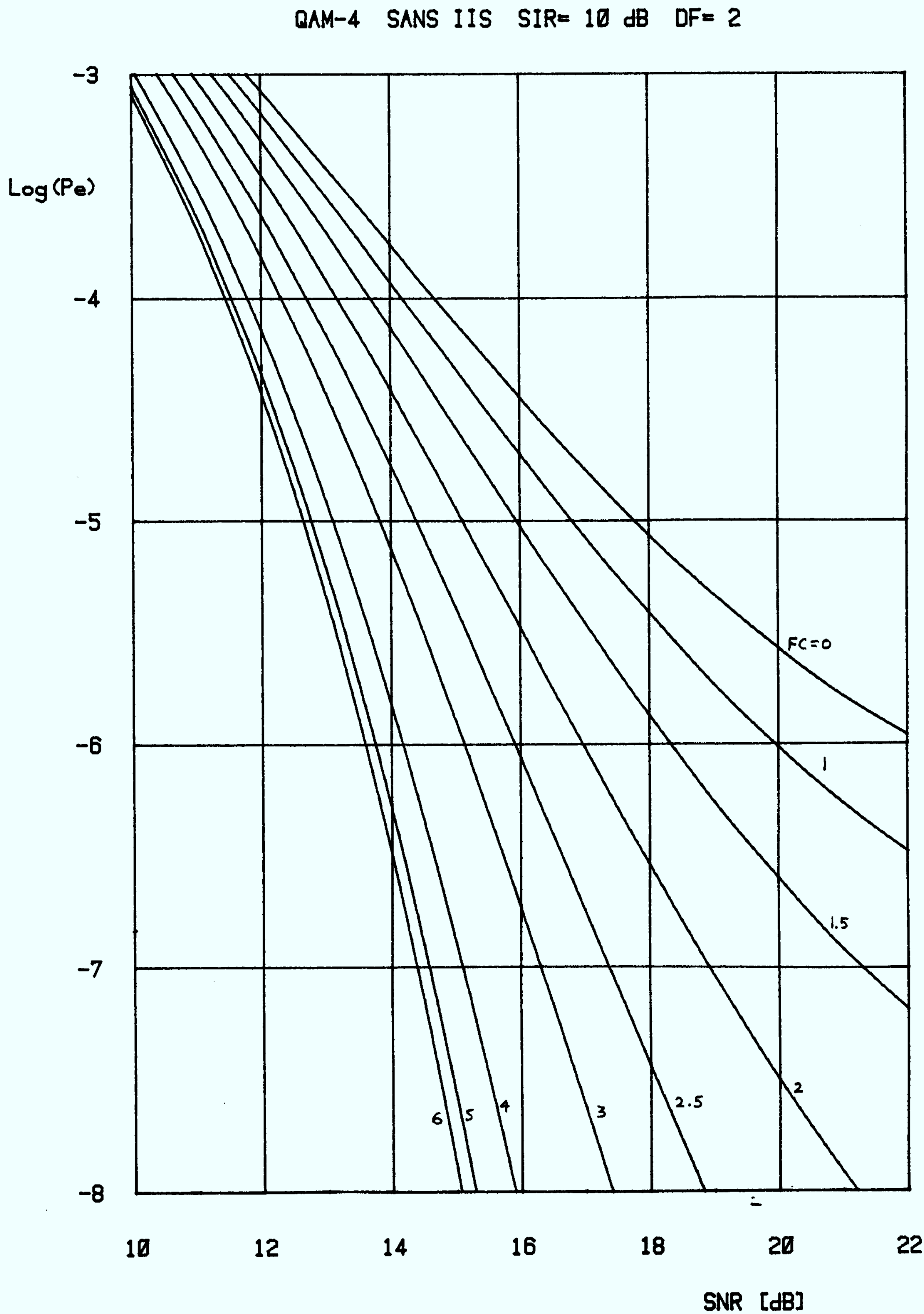

QAM-4 SANS IIS SIR= 10 dB DF= 2
Log(Pe)
-3
-4
-5
-6
-7
-8
10
12
14
16
18
20
22
SNR [dB]
FC=0
1
1.5
2
2.5
3
4
5
6

FIGURE A.3.11

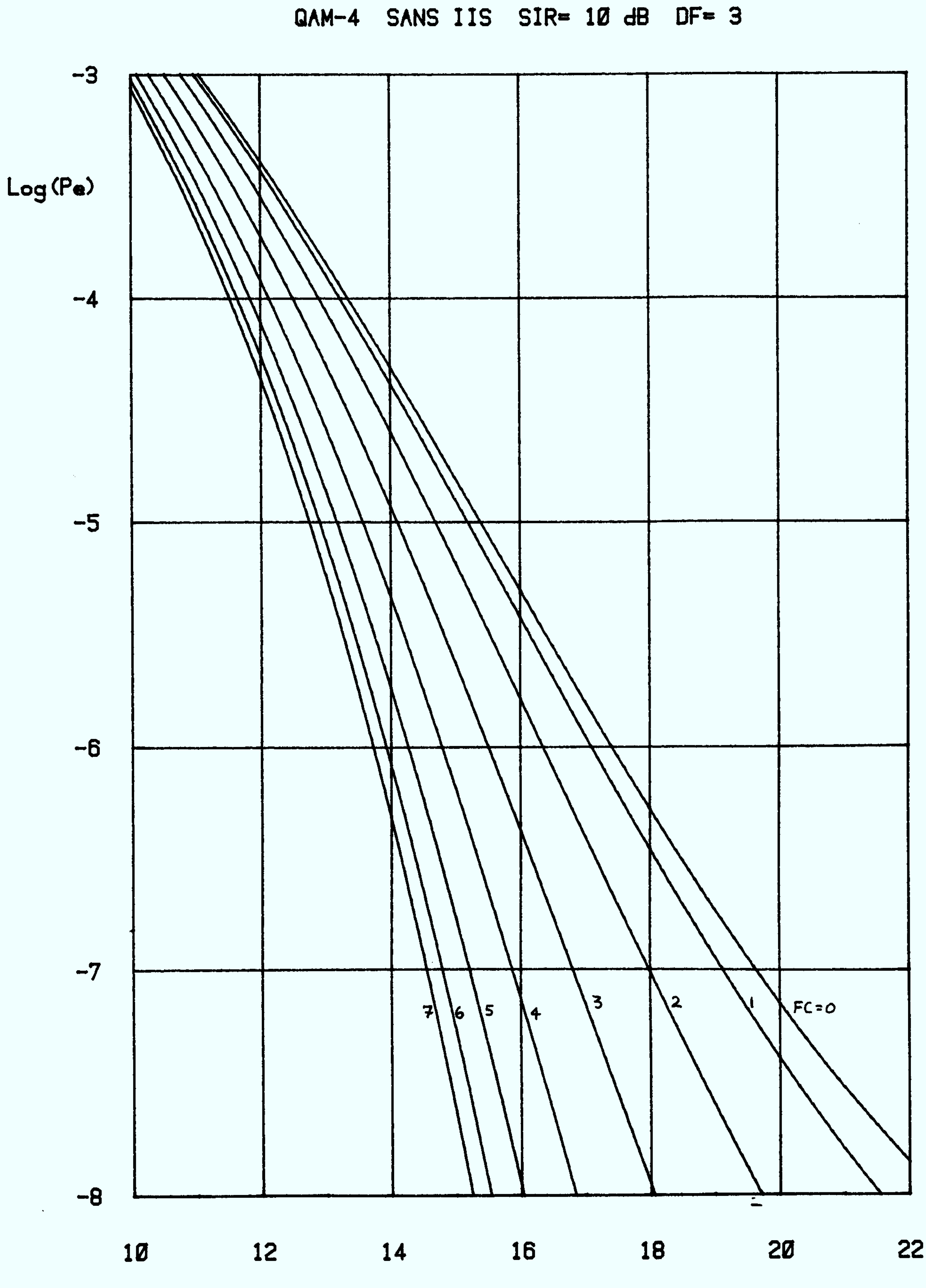

QAM-4 SANS IIS SIR= 10 dB DF= 3
Log(Pe)
-3
-4
-5
-6
-7
-8
7
6
5
4
3
2
1
FC=0
10
12
14
16
18
20
22
SNR [dB]

FIGURE A.3.12

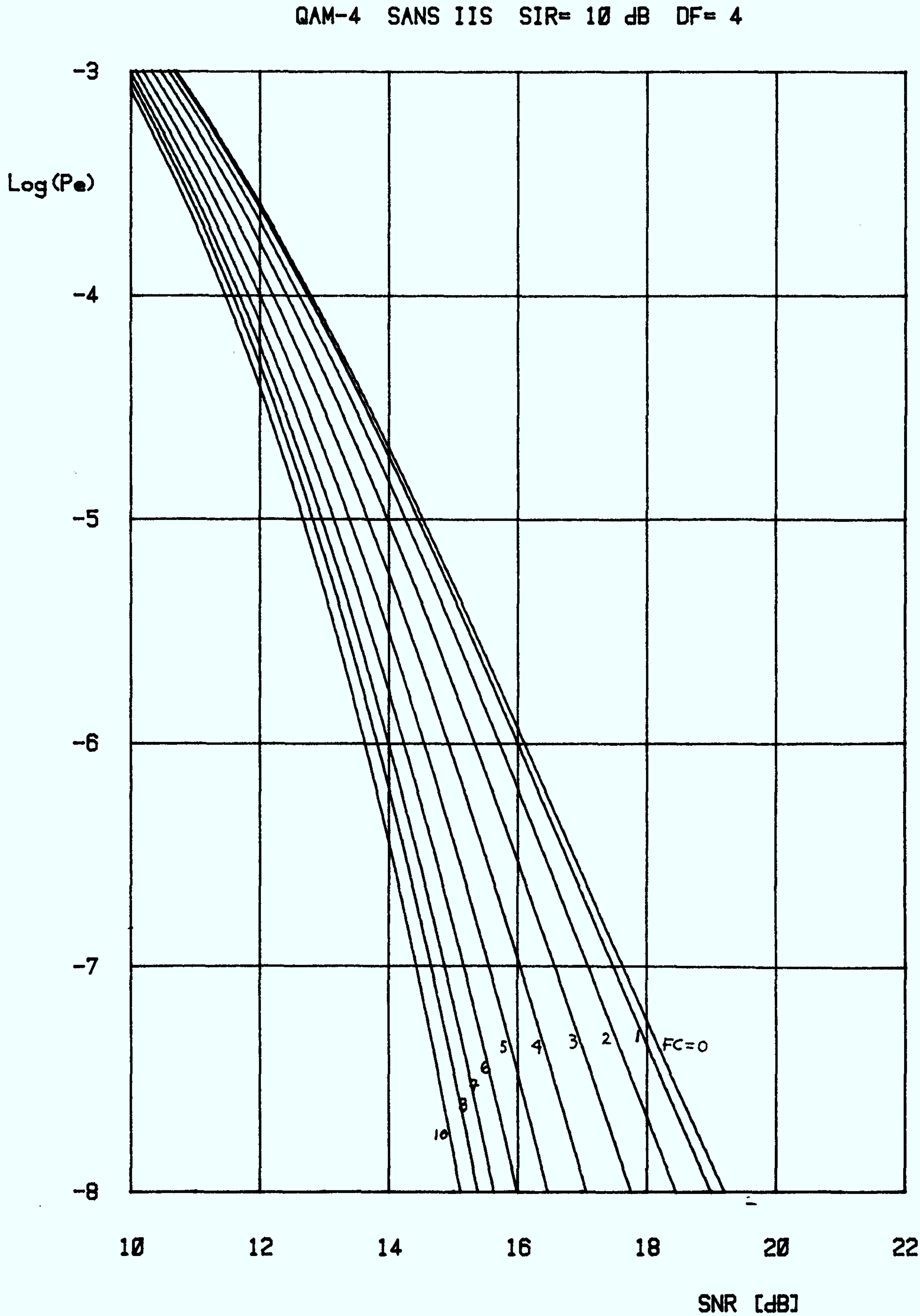

QAM-4 SANS IIS SIR= 10 dB DF= 4
Log(Pe)
-3
-4
-5
-6
-7
-8
10
12
14
16
18
20
22
SNR [dB]
FC=0
1
2
3
4
5
6
7
8
10

FIGURE A.3.13

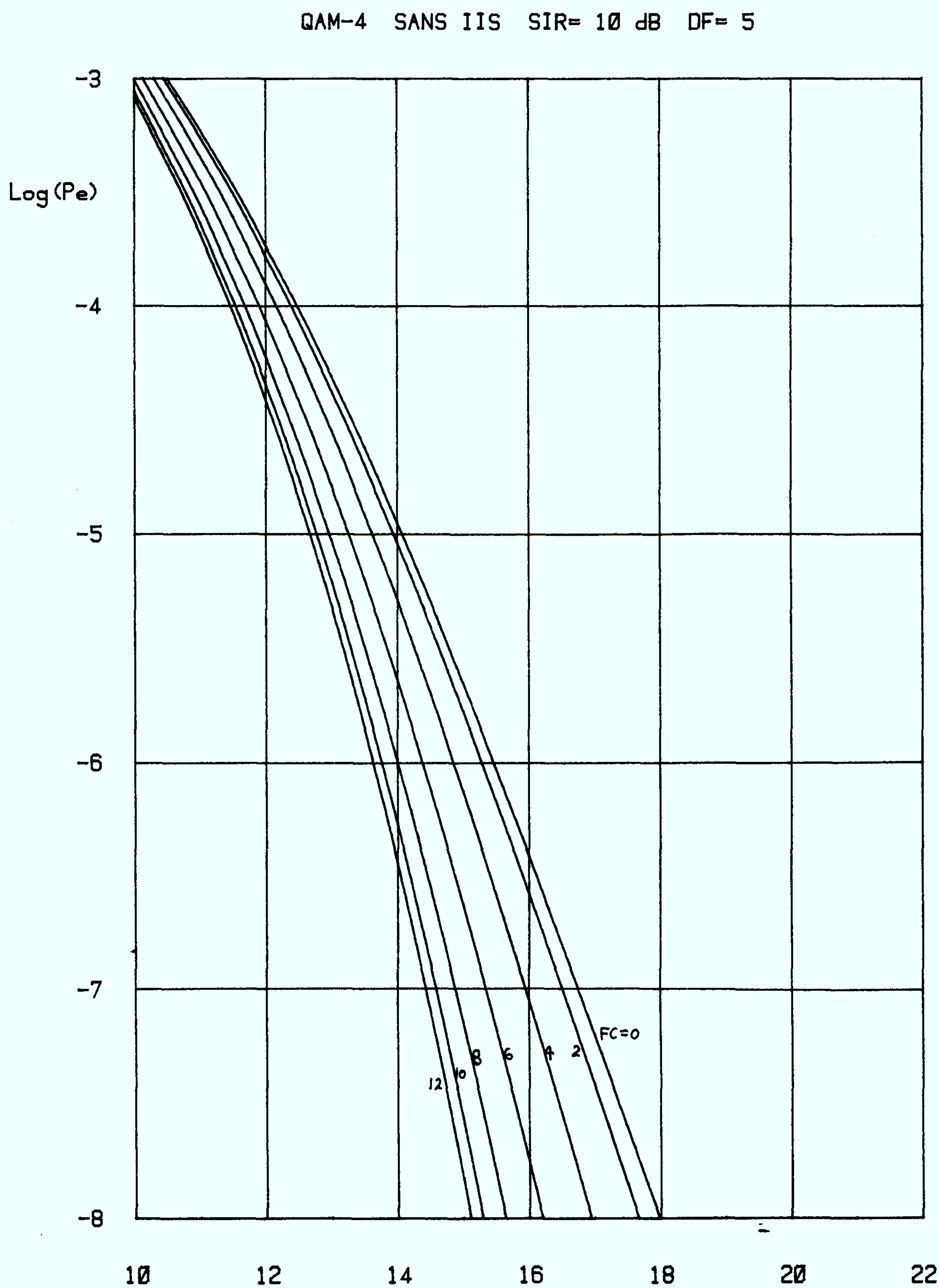

QAM-4 SANS IIS SIR= 10 dB DF= 5
Log(Pe)
-3
-4
-5
-6
-7
-8
FC=0
2
4
6
8
10
12
10
12
14
16
18
20
22
SNR [dB]

FIGURE A.3.14

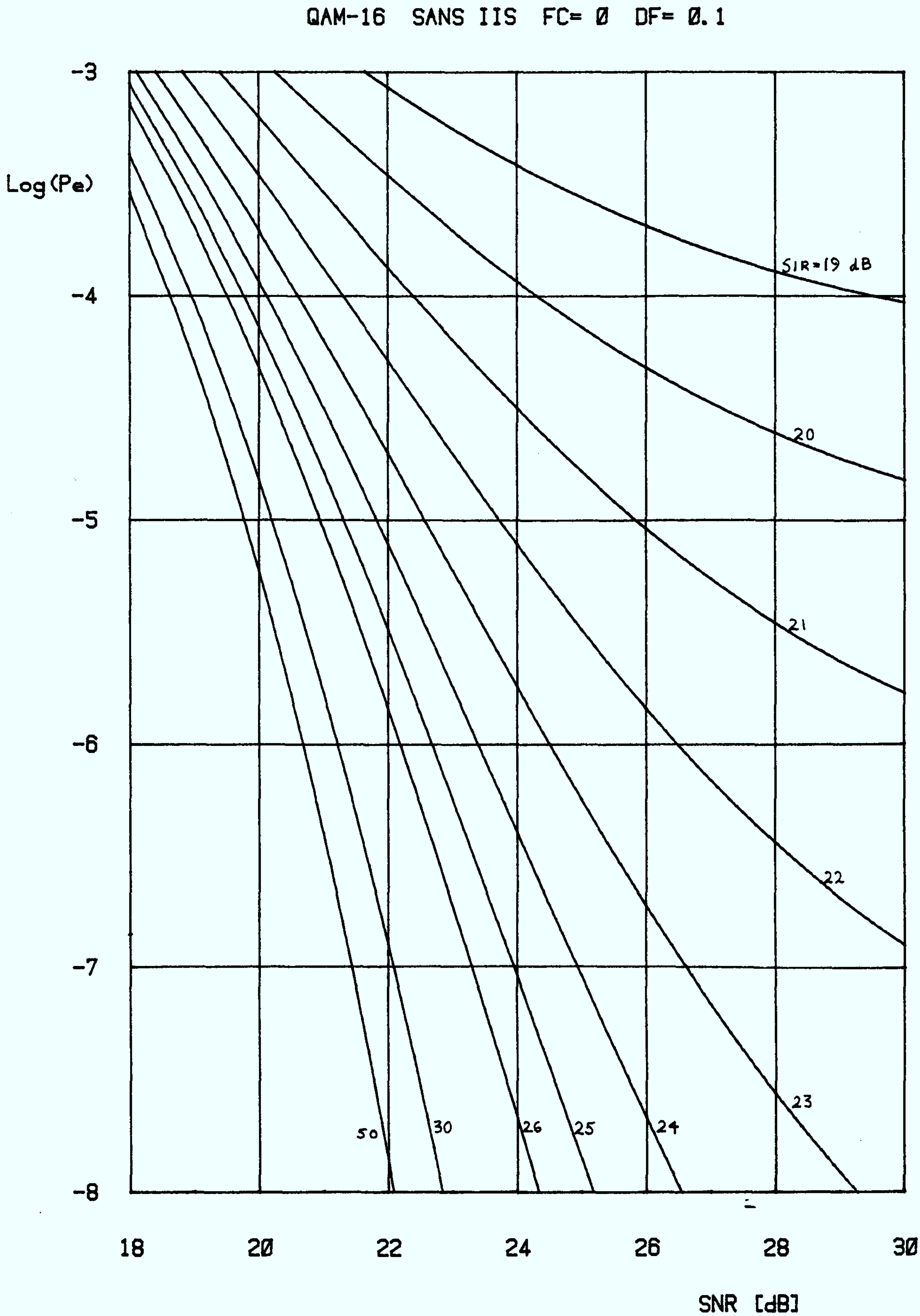

QAM-16 SANS IIS FC= Ø DF= Ø.1
Log(Pe)
-3
-4
-5
-6
-7
-8
SIR=19 dB
20
21
22
23
24
25
26
30
50
18
2Ø
22
24
26
28
3Ø
SNR [dB]

FIGURE A.3.15

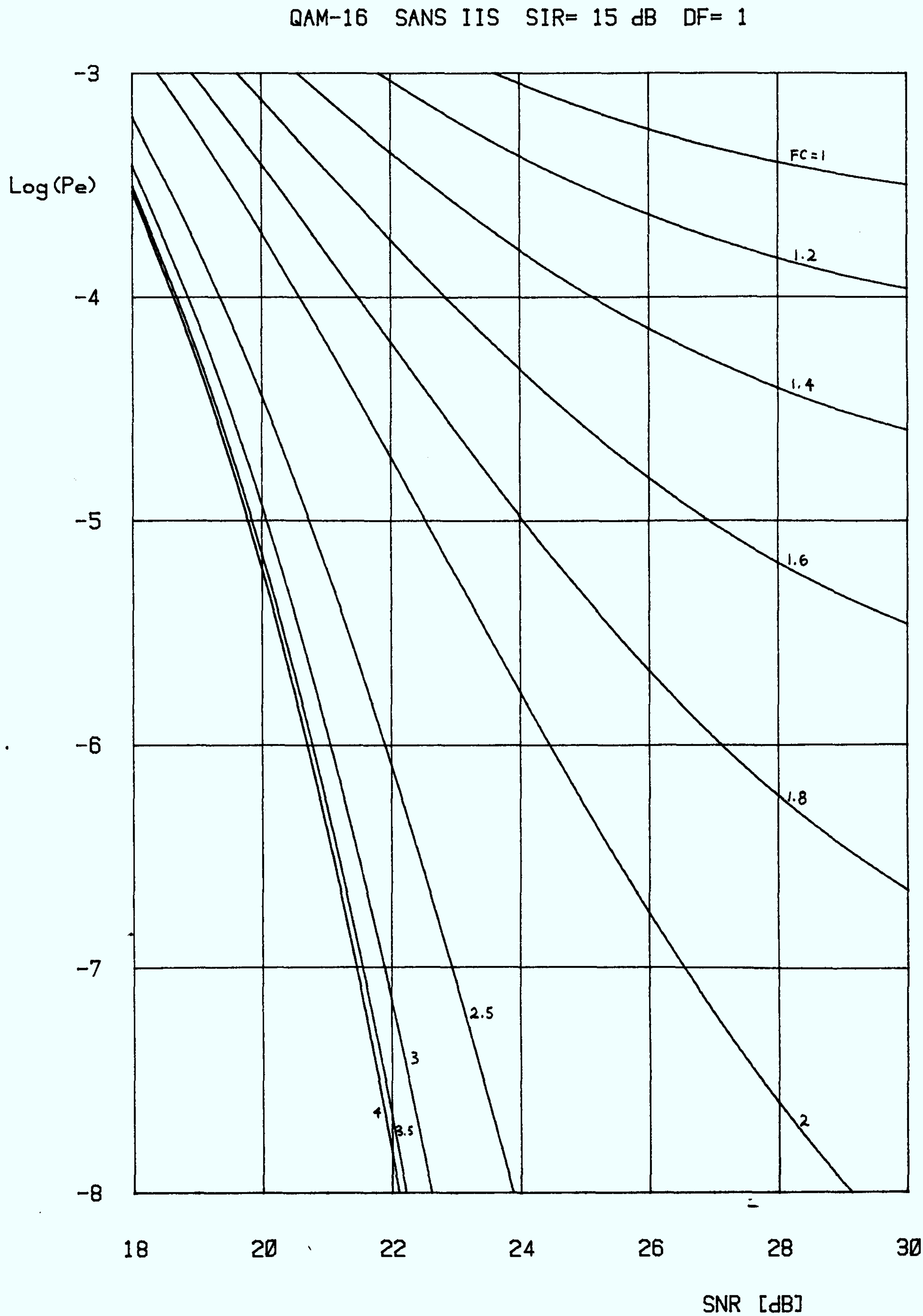

QAM-16 SANS IIS SIR= 15 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
18
20
22
24
26
28
30
SNR [dB]
FC=1
1.2
1.4
1.6
1.8
2
2.5
3
3.5
4

FIGURE A.3.16

QAM-16 SANS IIS SIR= 15 dB DF= 2

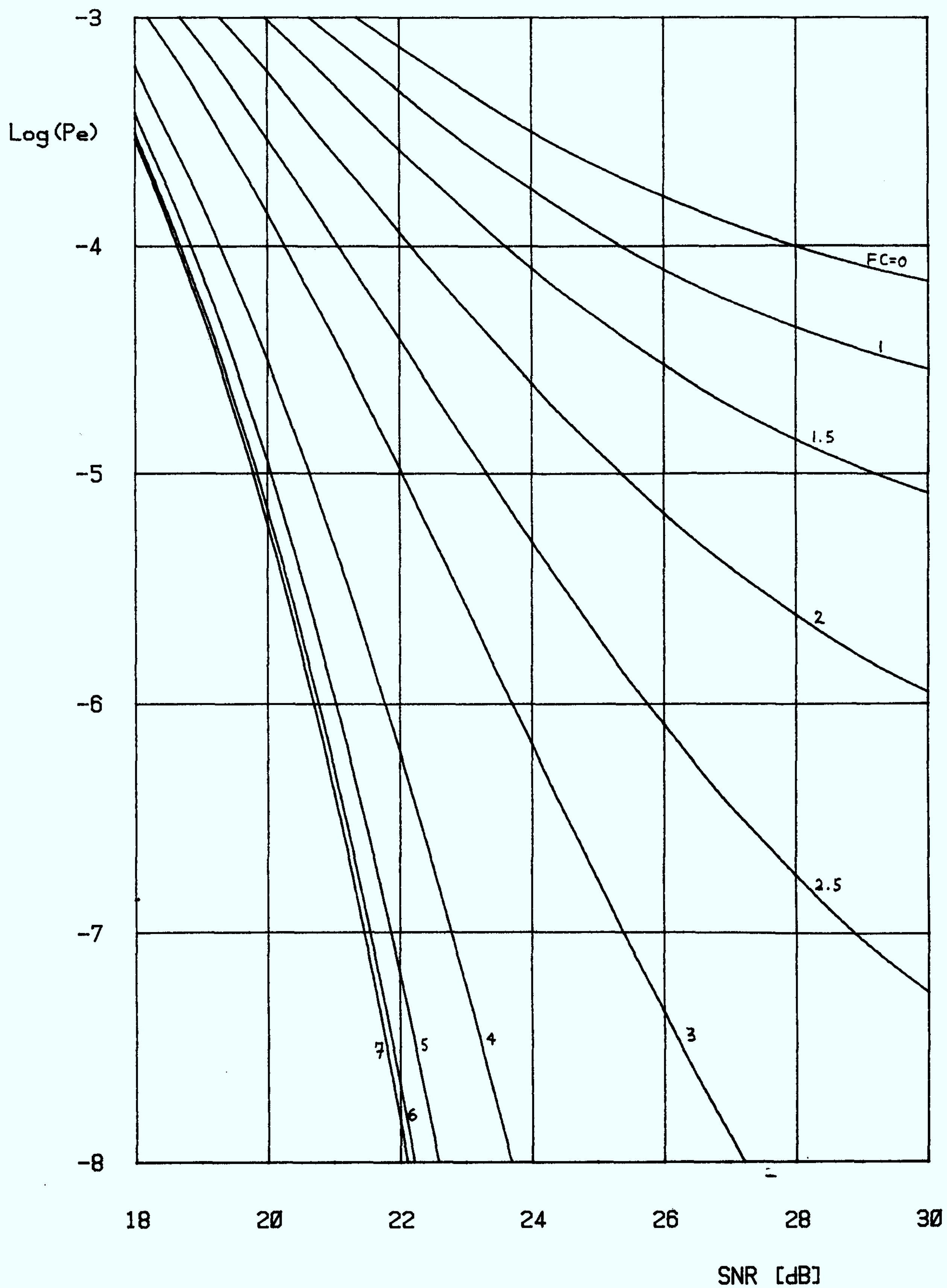


FIGURE A.3.17

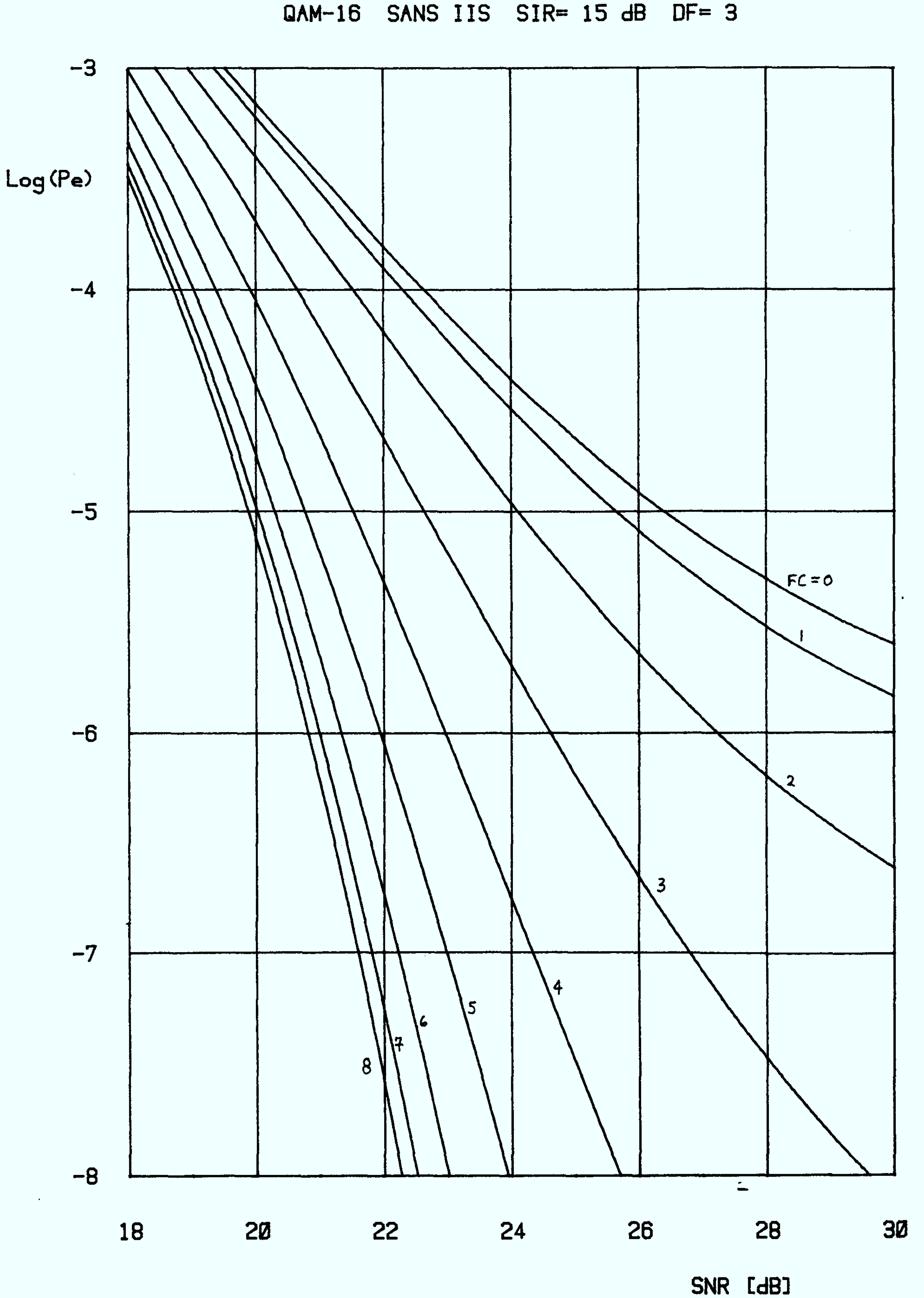

QAM-16 SANS IIS SIR= 15 dB DF= 3
Log(Pe)
-3
-4
-5
-6
-7
-8
18
20
22
24
26
28
30
SNR [dB]
FC=0
1
2
3
4
5
6
7
8

FIGURE A.3.18

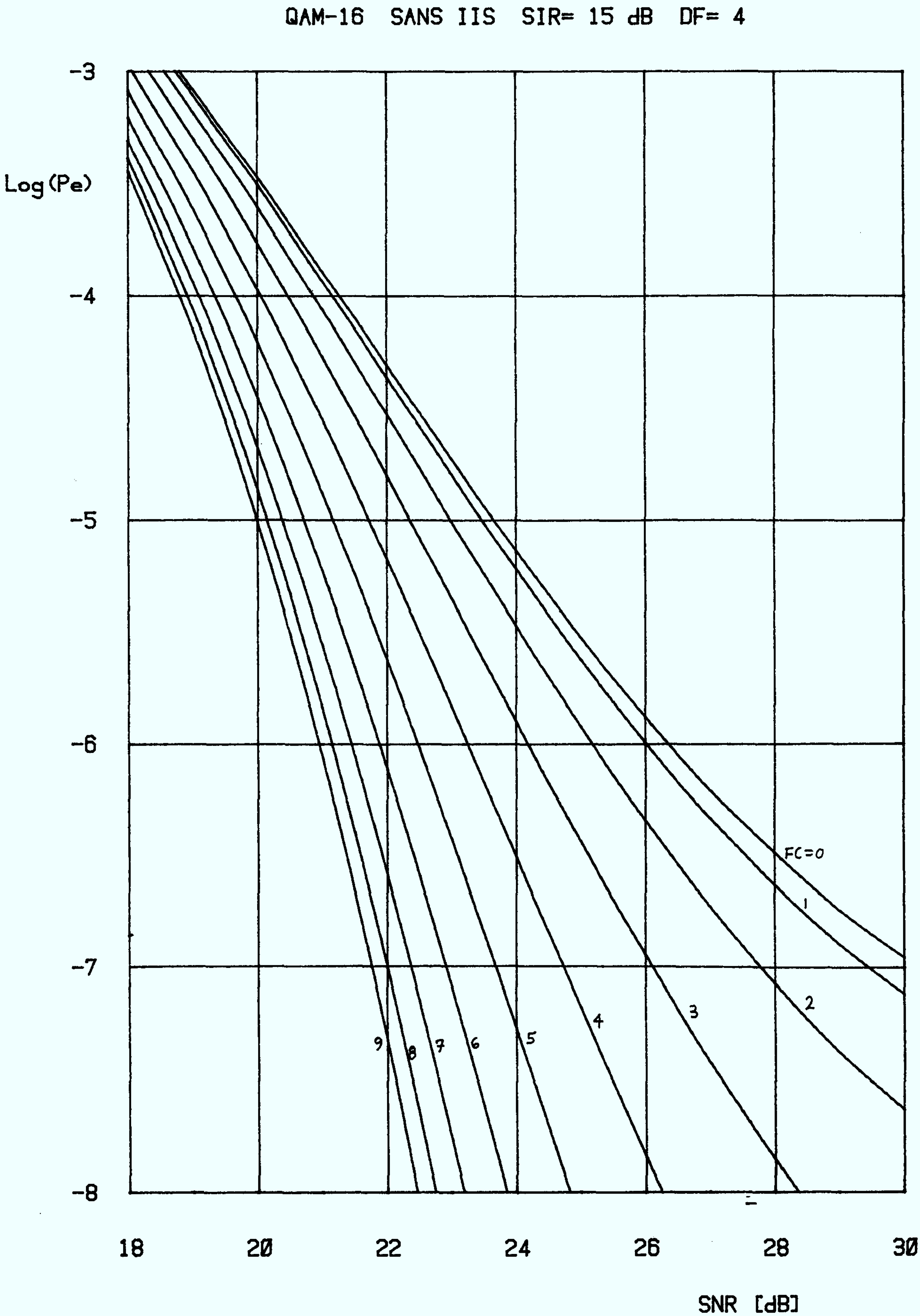

QAM-16 SANS IIS SIR= 15 dB DF= 4
Log(Pe)
-3
-4
-5
-6
-7
-8
18
20
22
24
26
28
30
SNR [dB]
FC=0
1
2
3
4
5
6
7
8
9

FIGURE A.3.19

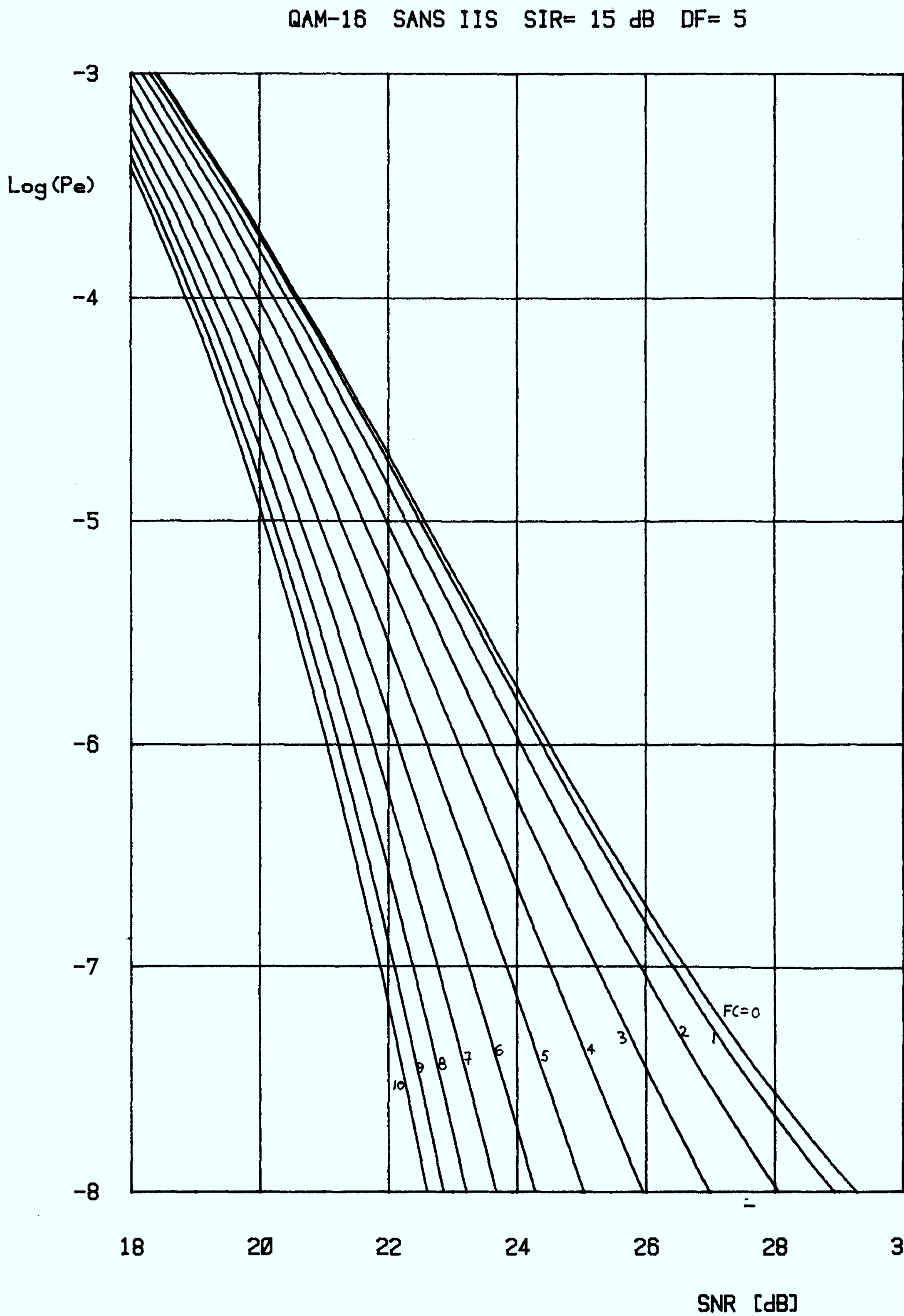

QAM-16 SANS IIS SIR= 15 dB DF= 5
Log(Pe)
-3
-4
-5
-6
-7
-8
18
20
22
24
26
28
30
SNR [dB]
FC=0
1
2
3
4
5
6
7
8
9
10

FIGURE A.3.20

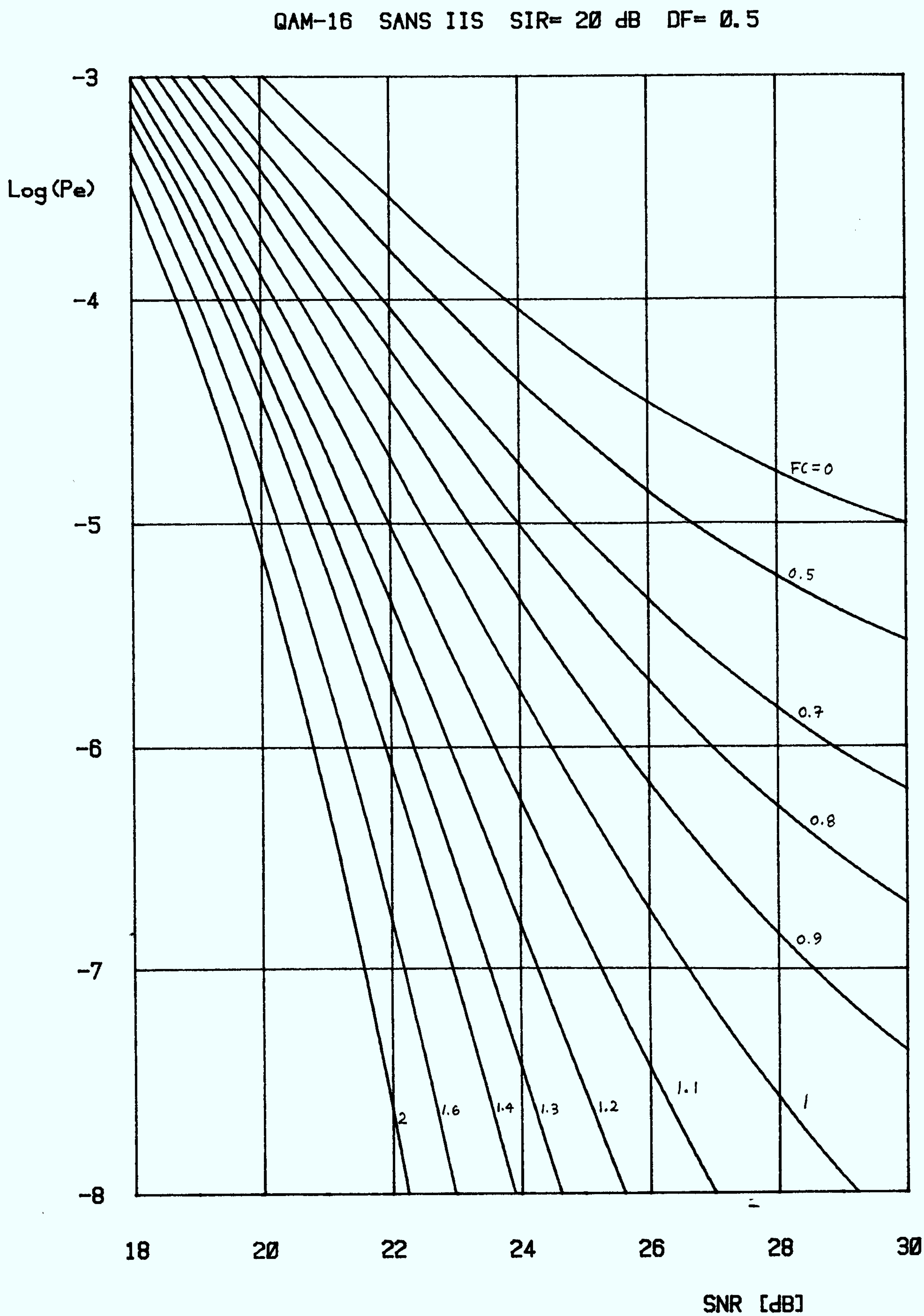

QAM-16 SANS IIS SIR= 20 dB DF= 0.5
Log (Pe)
-3
-4
-5
-6
-7
-8
FC=0
0.5
0.7
0.8
0.9
1
1.1
1.2
1.3
1.4
1.6
2
18
20
22
24
26
28
30
SNR [dB]

FIGURE A.3.21

QAM-16 SANS IIS SIR= 20 dB DF= 1

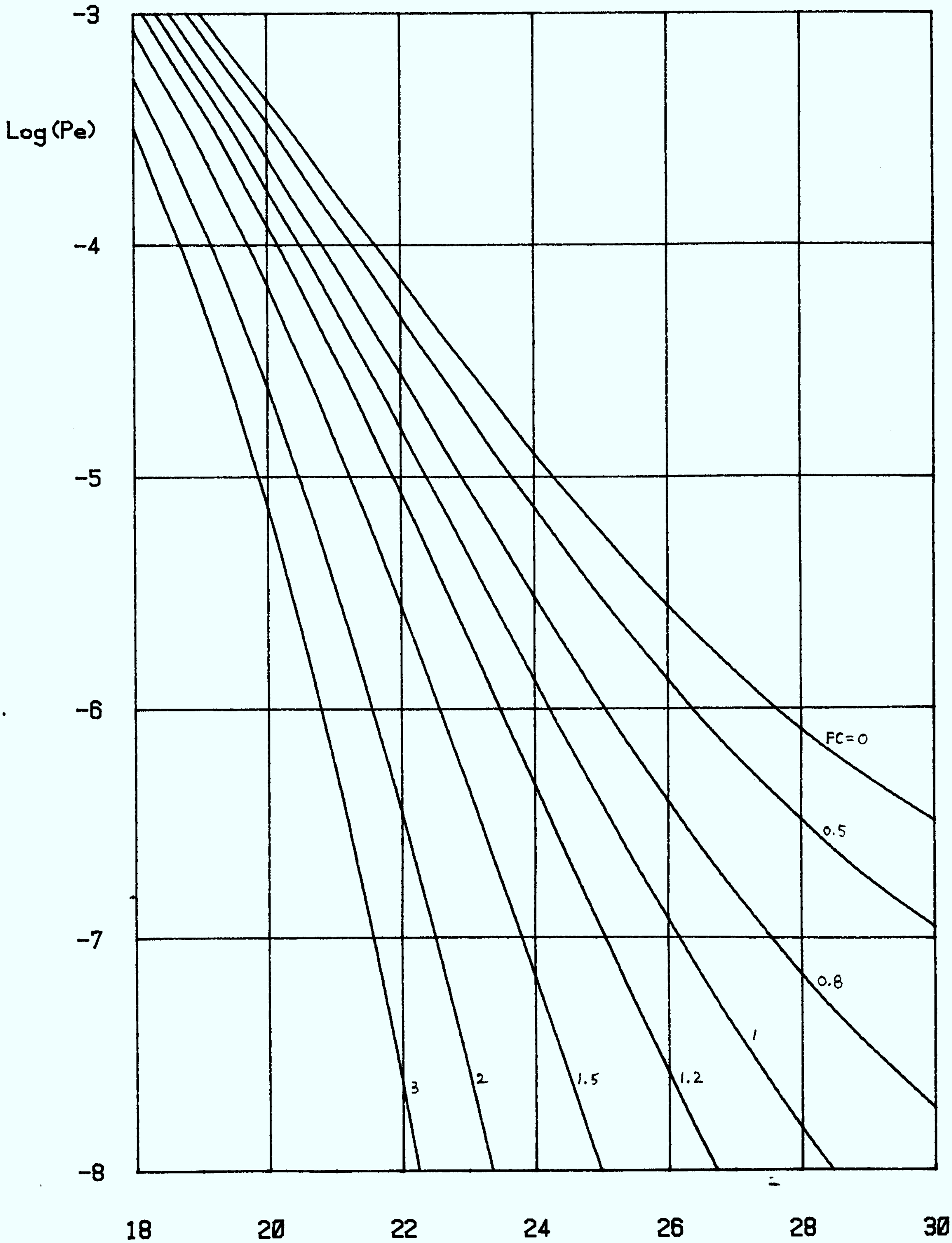


FIGURE A.3.22

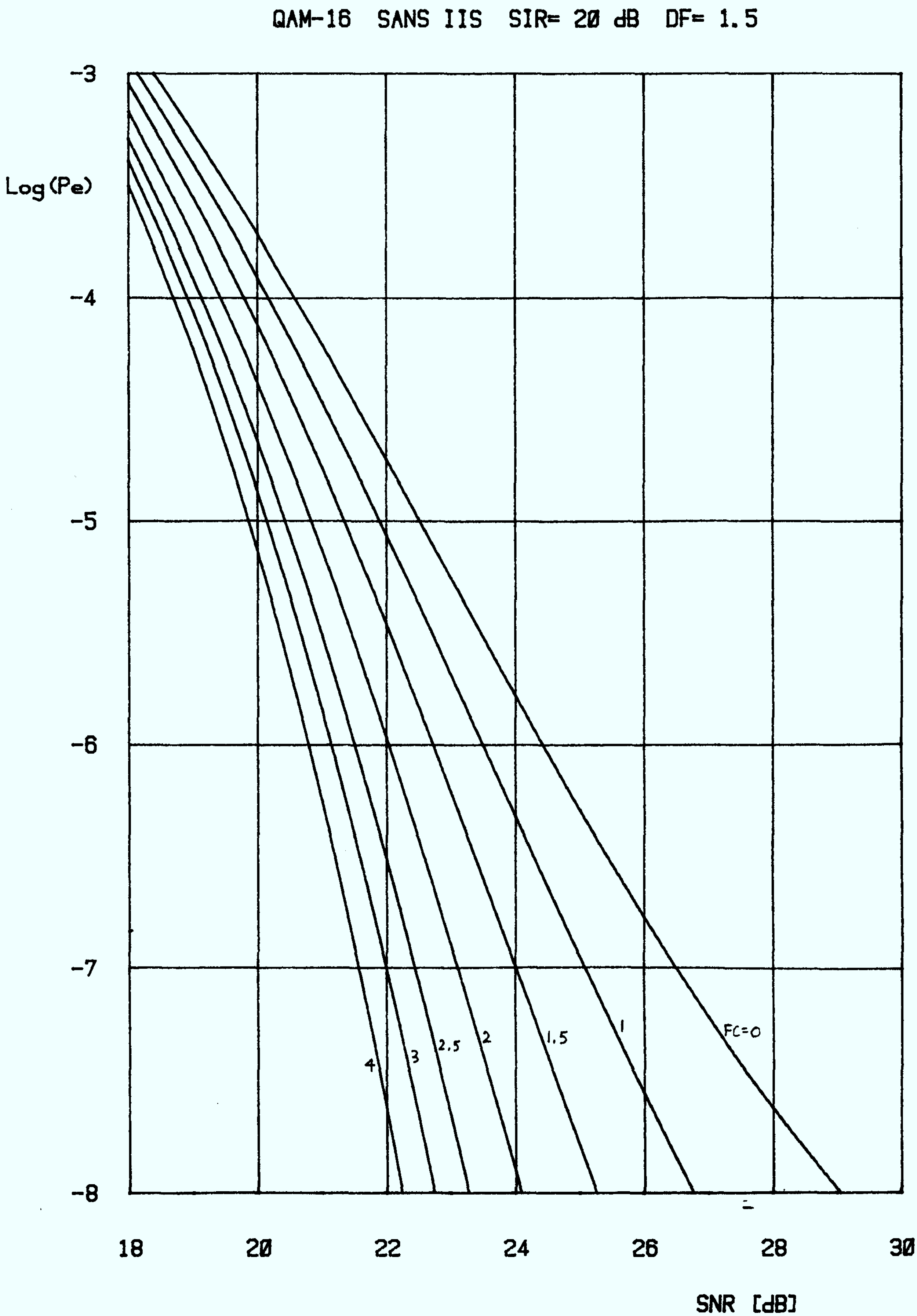

QAM-16 SANS IIS SIR= 20 dB DF= 1.5
Log(Pe)
-3
-4
-5
-6
-7
-8
18
20
22
24
26
28
30
SNR [dB]
4
3
2.5
2
1.5
1
FC=0

FIGURE A.3.23

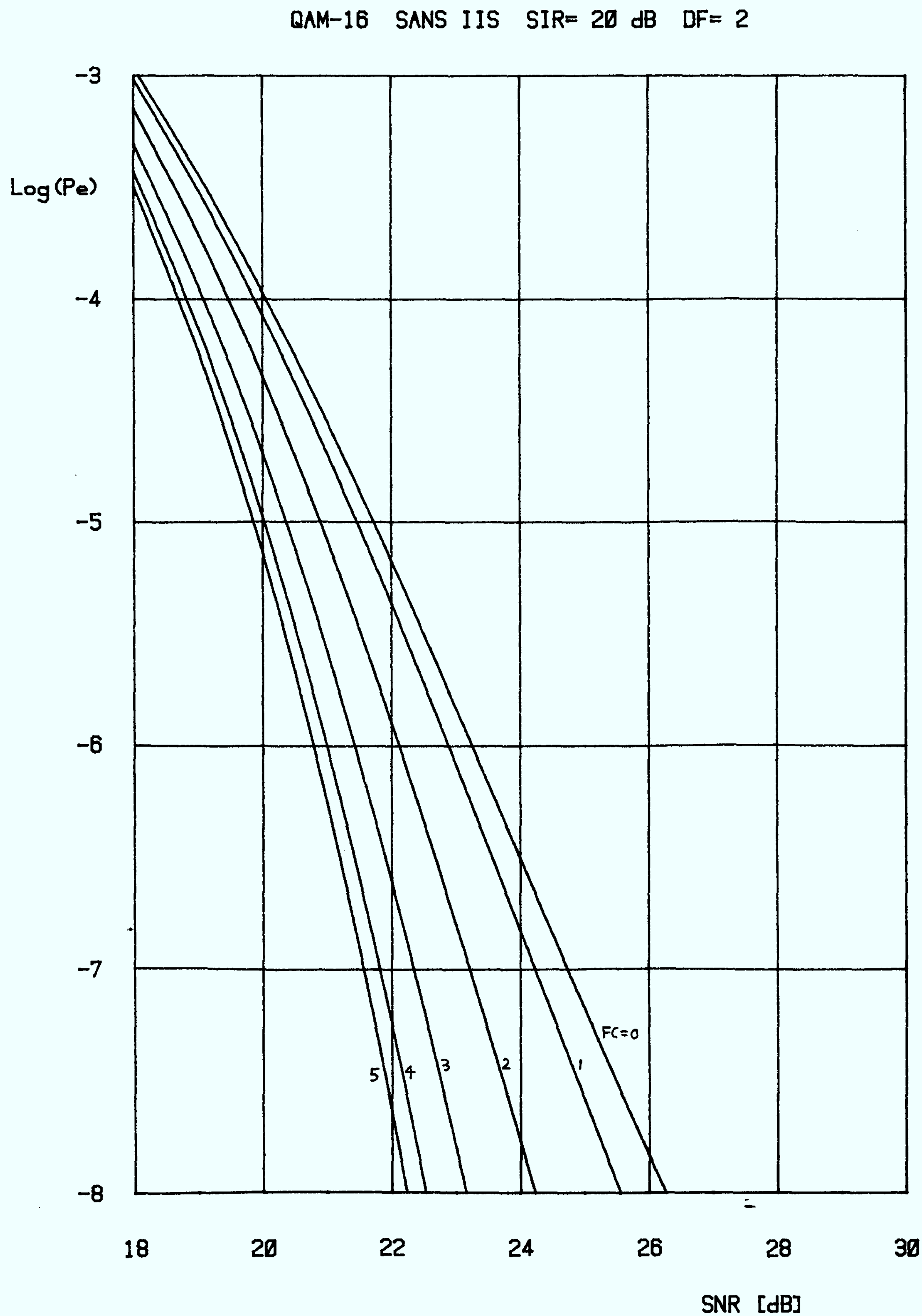

QAM-16 SANS IIS SIR= 20 dB DF= 2
Log(Pe)
-3
-4
-5
-6
-7
-8
18
20
22
24
26
28
30
SNR [dB]
FC=0
1
2
3
4
5

FIGURE A.3.24

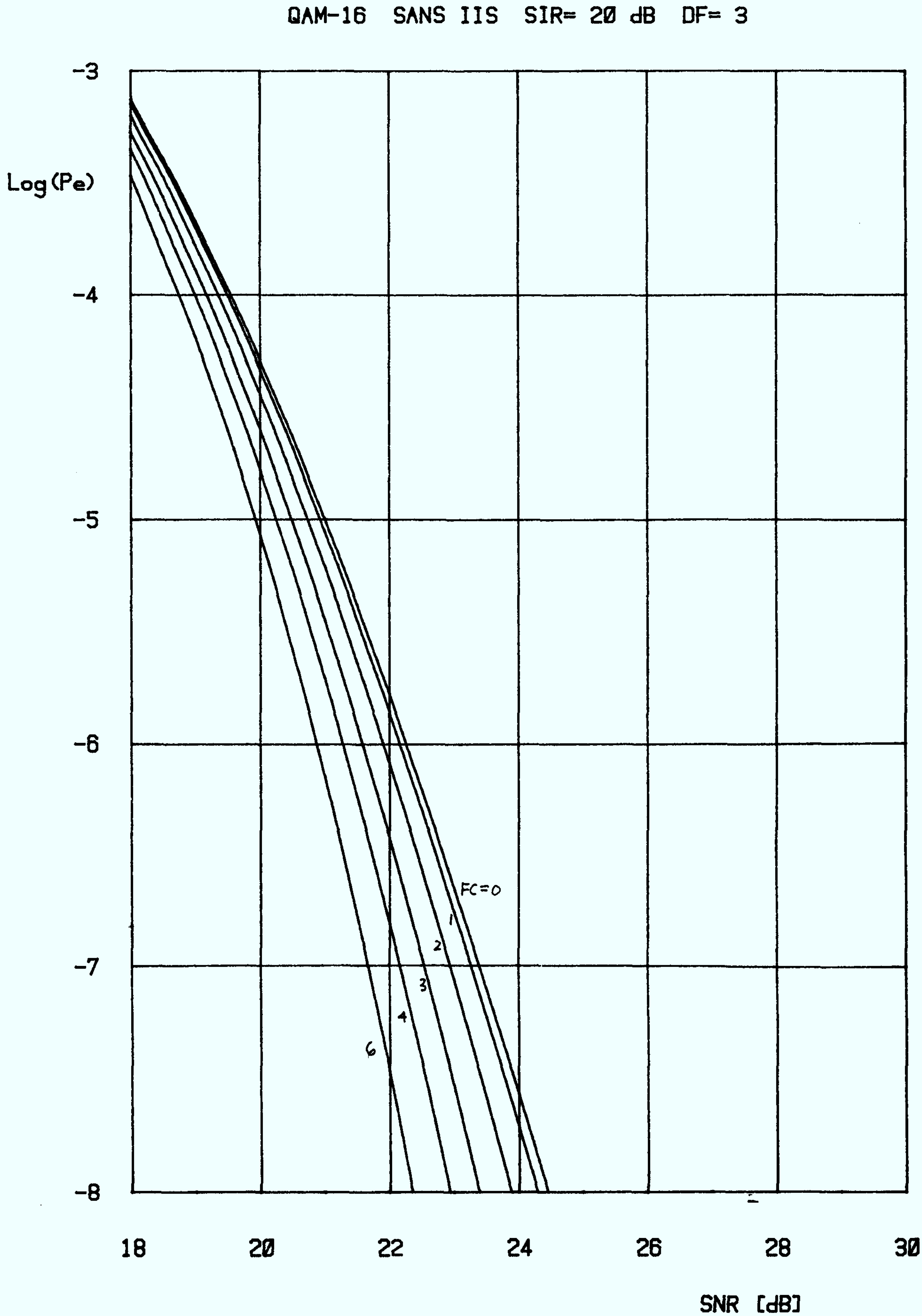

QAM-16 SANS IIS SIR= 20 dB DF= 3
Log(Pe)
-3
-4
-5
-6
-7
-8
FC=0
1
2
3
4
6
18
20
22
24
26
28
30
SNR [dB]

FIGURE A.3.25

QAM-16 SANS IIS SIR= 25 dB DF= Ø.1

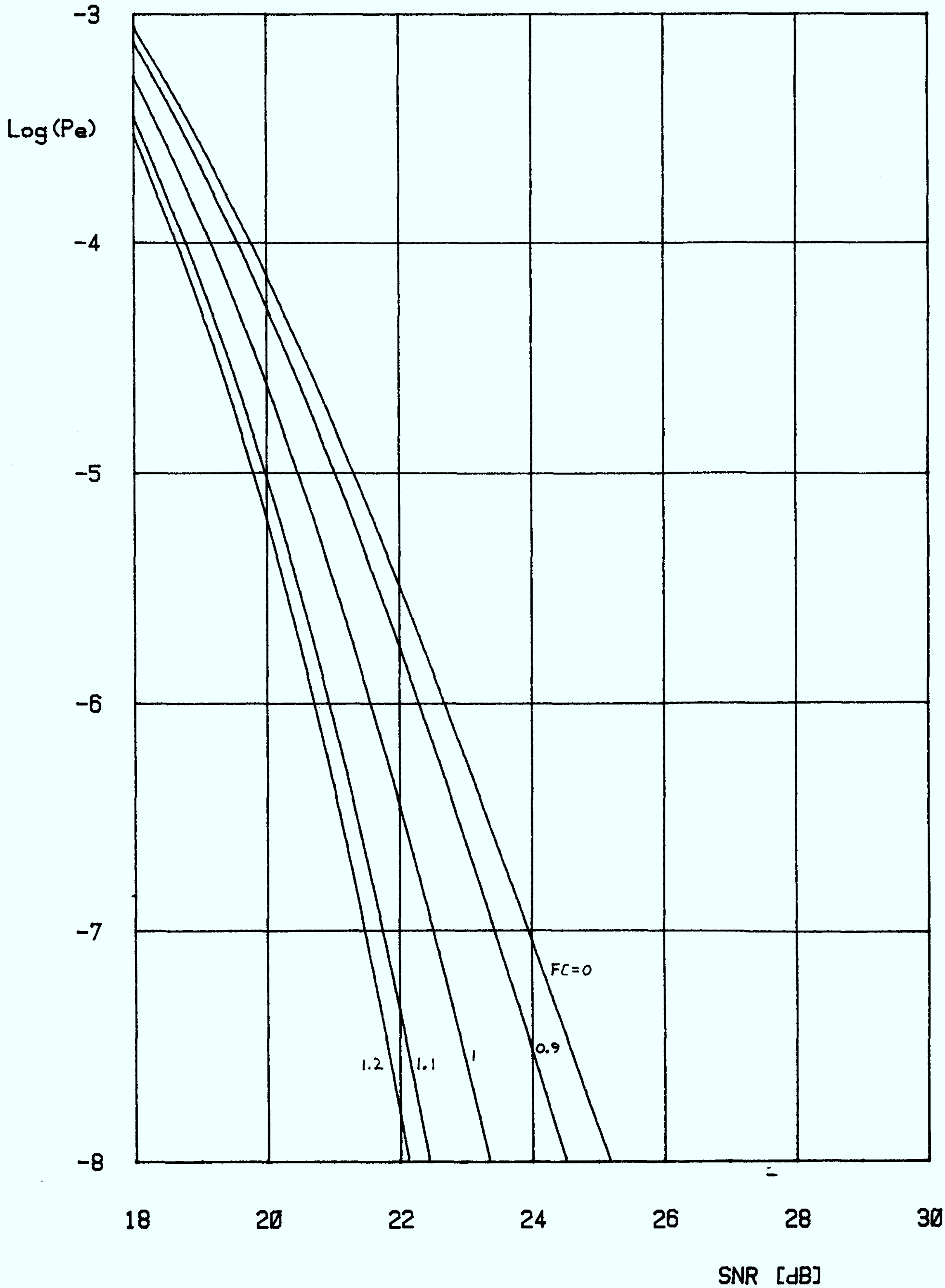


FIGURE A.3.26

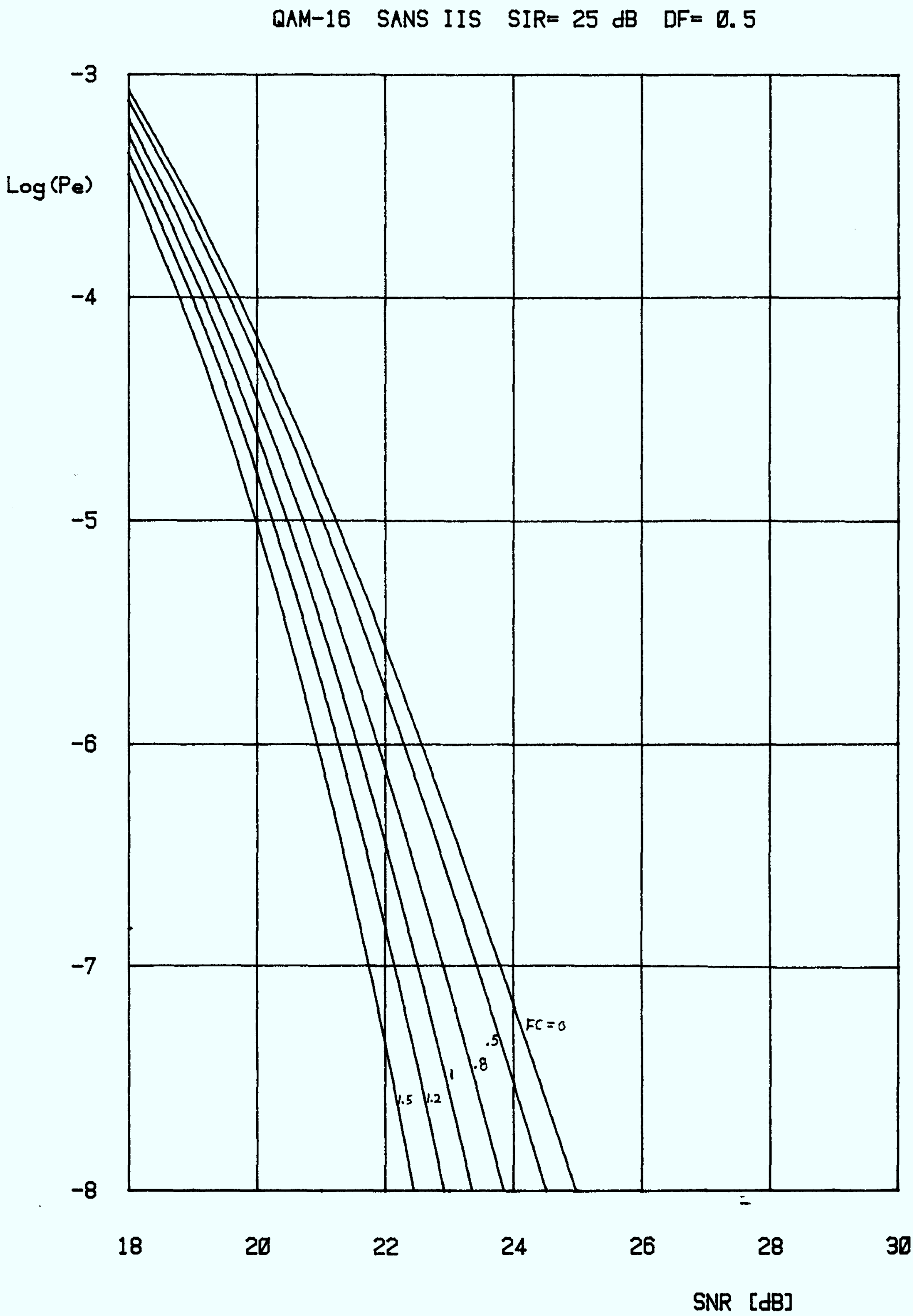

QAM-16 SANS IIS SIR= 25 dB DF= Ø.5
Log (Pe)
-3
-4
-5
-6
-7
-8
18
2Ø
22
24
26
28
3Ø
SNR [dB]
FC = 0
.5
.8
1
1.2
1.5

FIGURE A.3.27

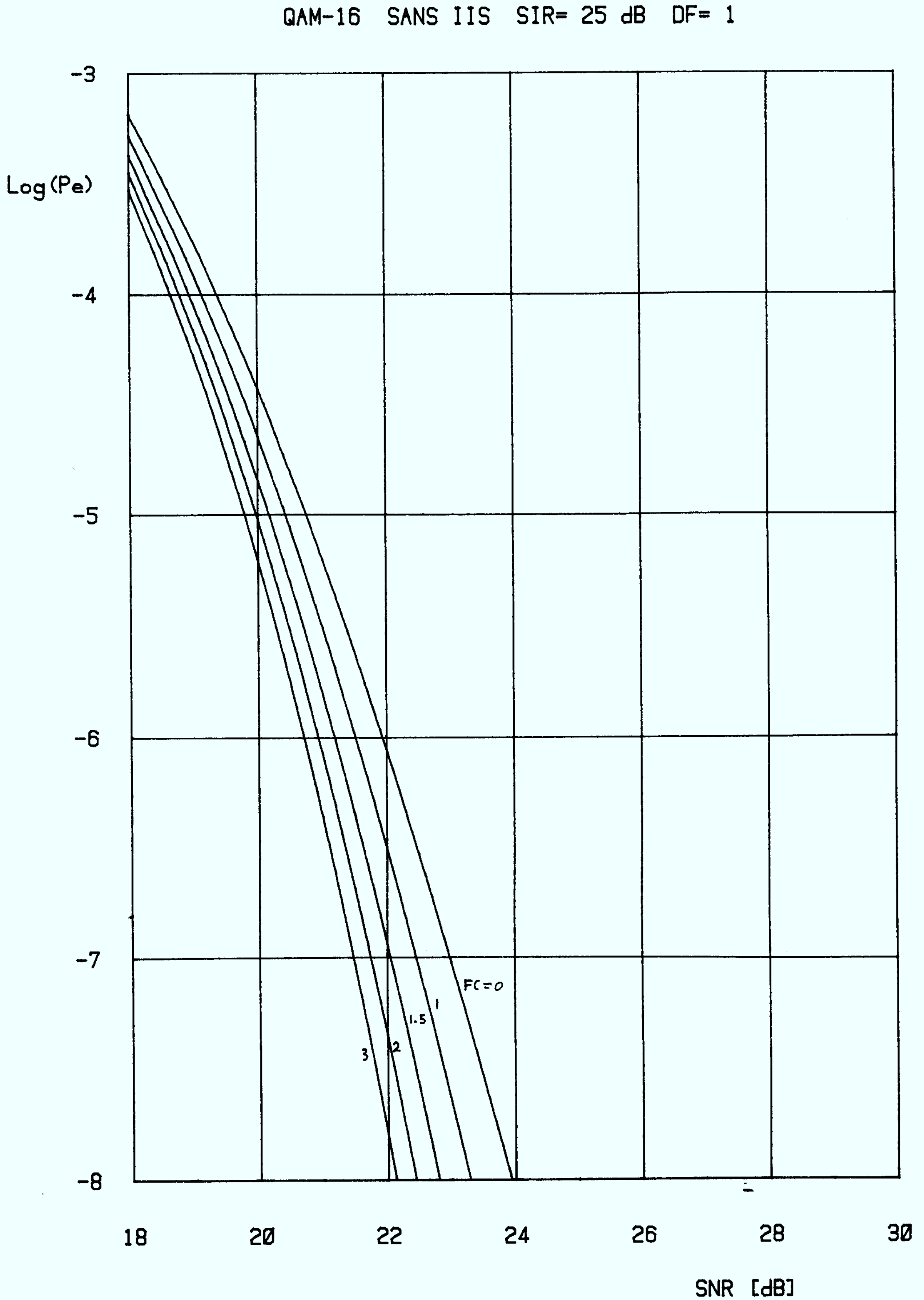

QAM-16 SANS IIS SIR= 25 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
18
20
22
24
26
28
30
SNR [dB]
FC=0
1
1.5
2
3

FIGURE A.3.28

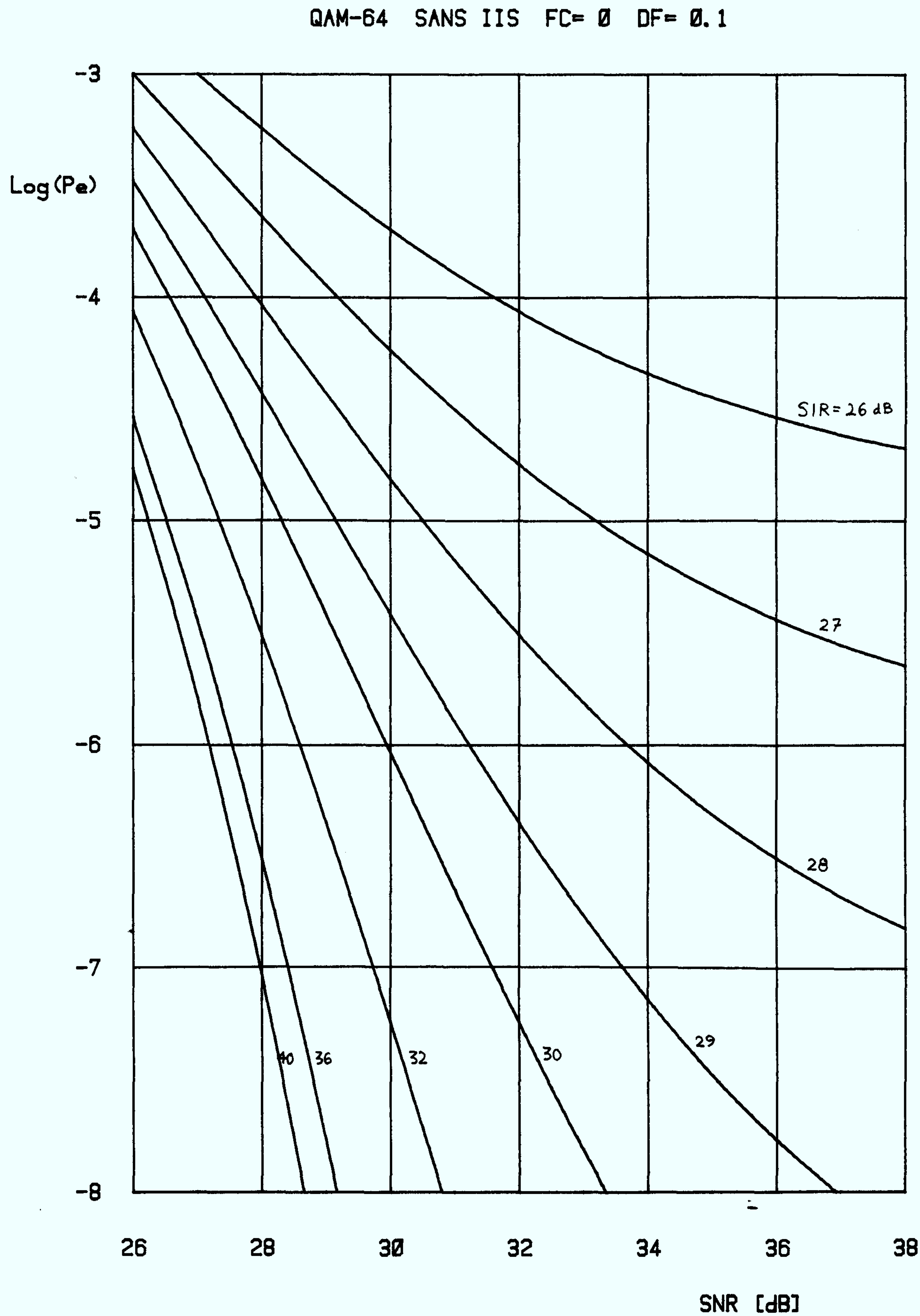

QAM-64 SANS IIS FC= Ø DF= Ø.1
Log(Pe)
-3
-4
-5
-6
-7
-8
26
28
3Ø
32
34
36
38
SNR [dB]
SIR= 26 dB
27
28
29
30
32
36
40

FIGURE A.3.29

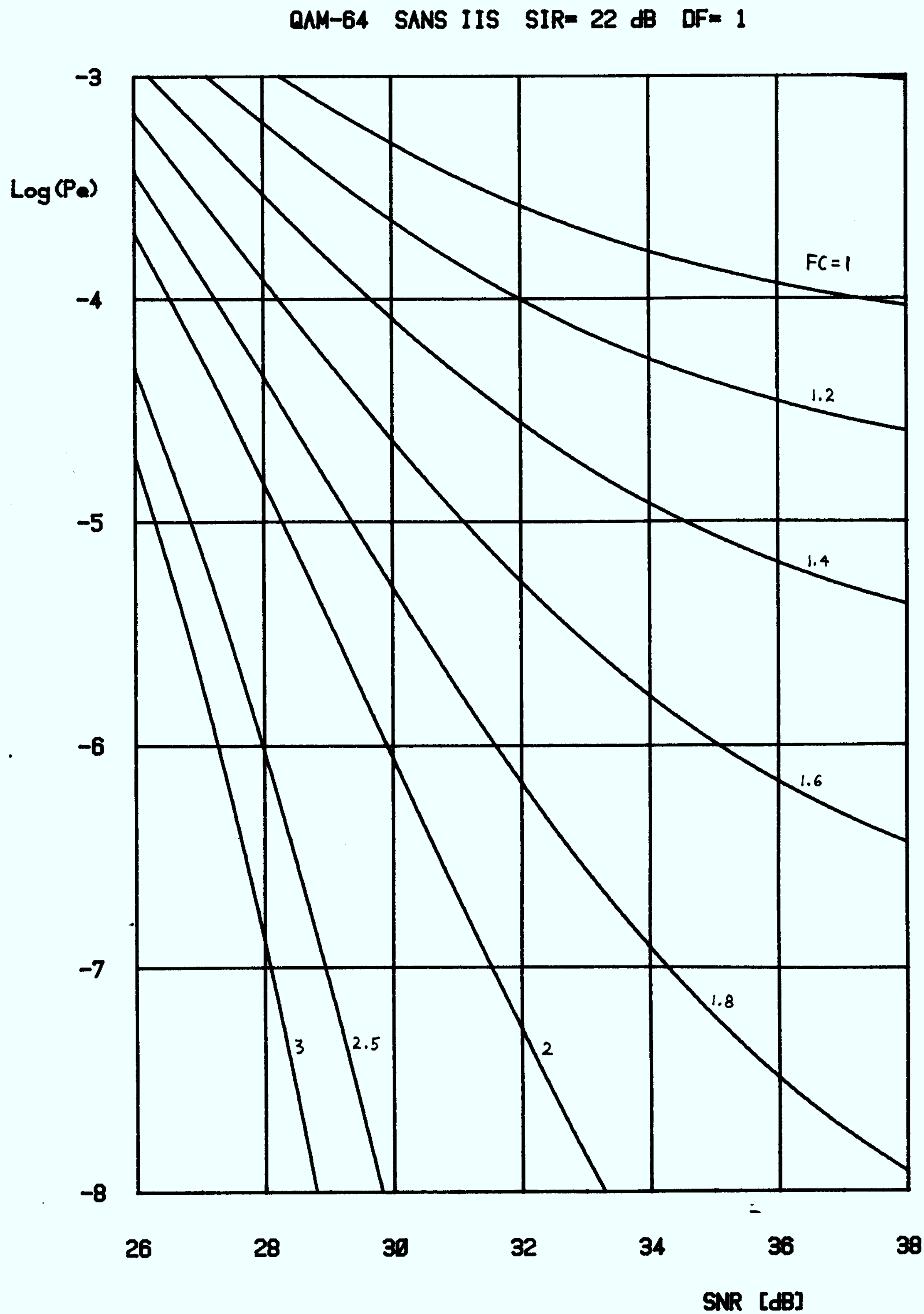

QAM-64 SANS IIS SIR= 22 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
FC=1
1.2
1.4
1.6
1.8
2
2.5
3
26
28
30
32
34
36
38
SNR [dB]

FIGURE A.3.30

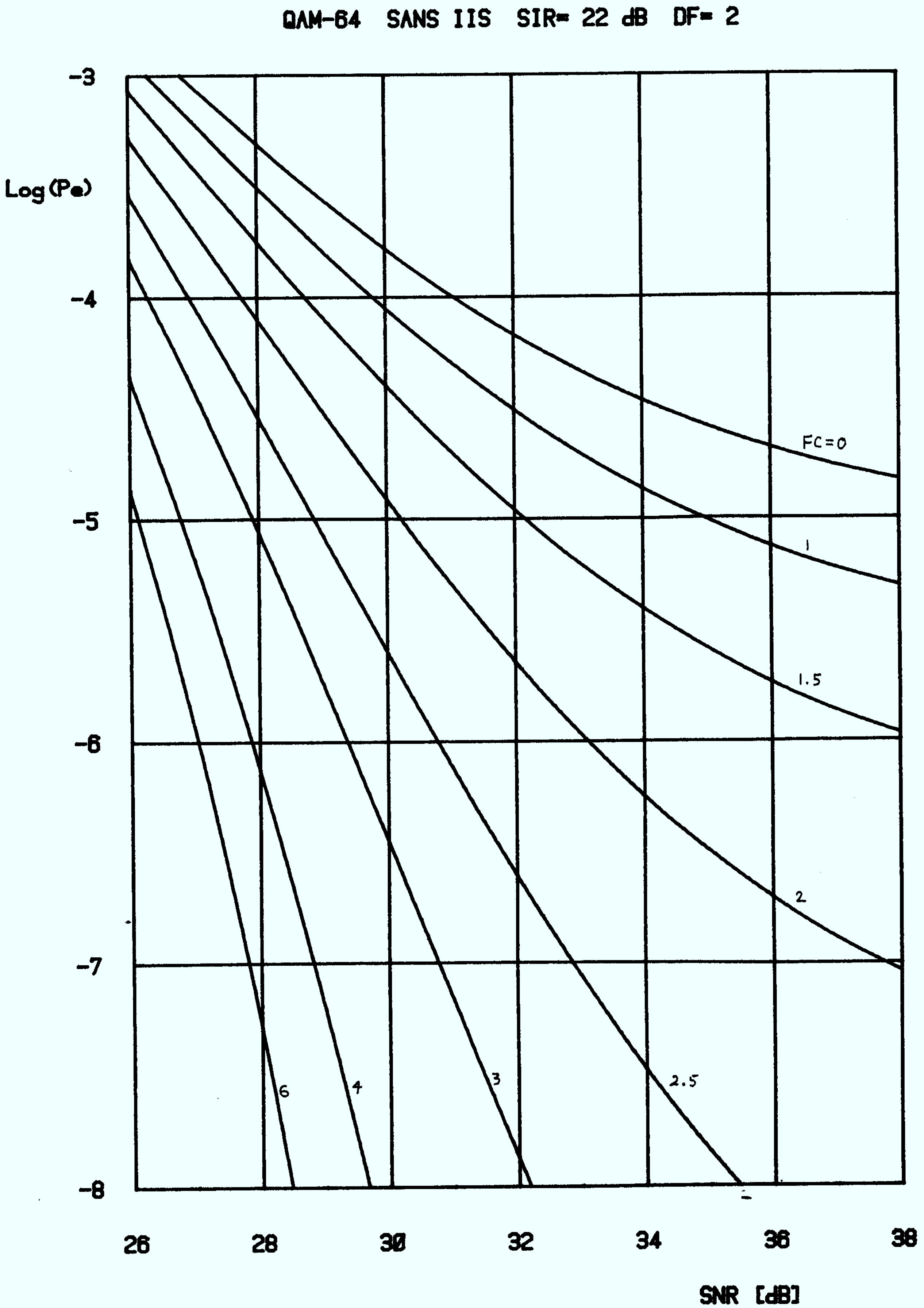

QAM-64 SANS IIS SIR= 22 dB DF= 2
Log(Pe)
-3
-4
-5
-6
-7
-8
26
28
30
32
34
36
38
SNR [dB]
FC=0
1
1.5
2
2.5
3
4
6

FIGURE A.3.31

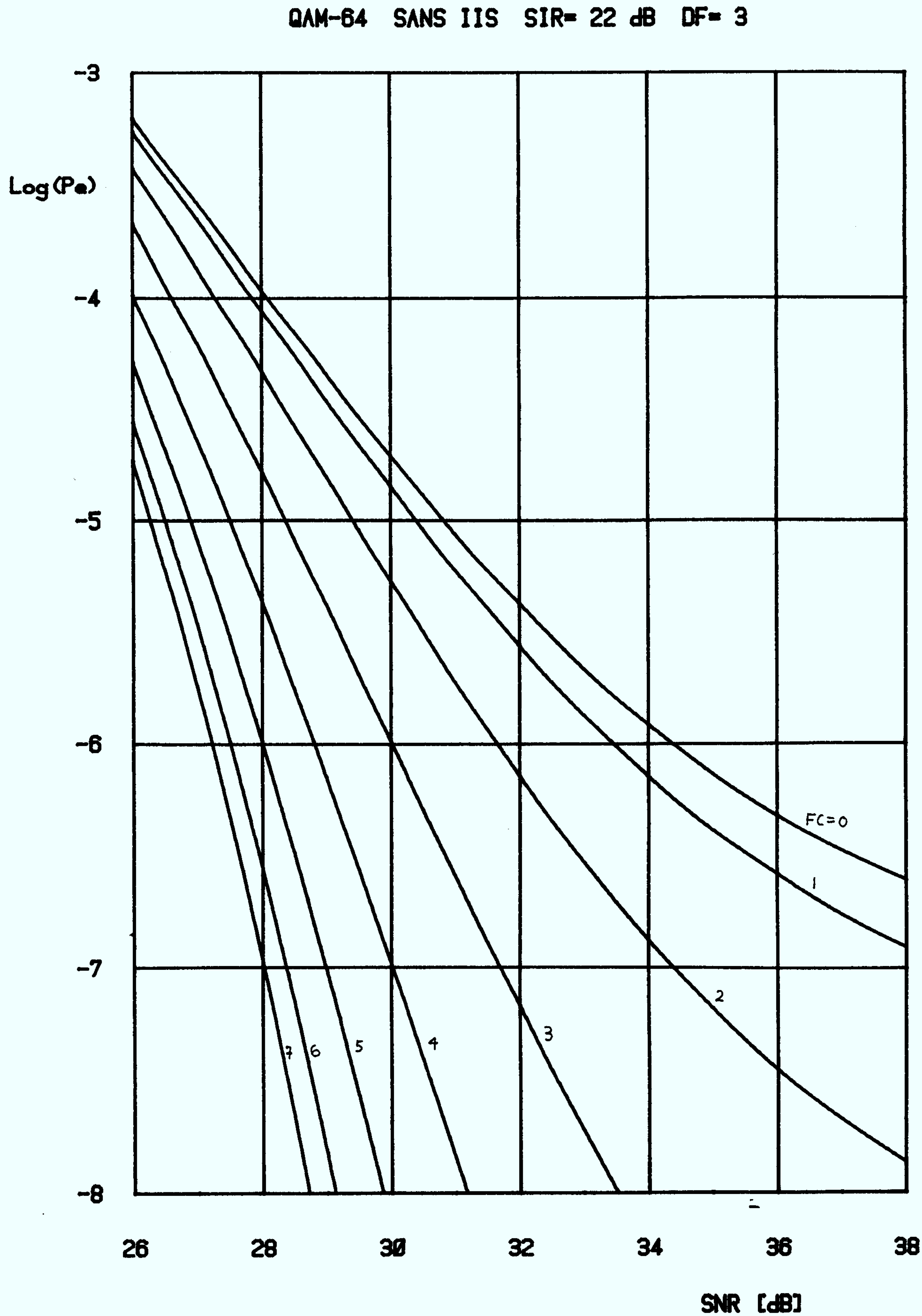

QAM-64 SANS IIS SIR= 22 dB DF= 3
Log(Pe)
-3
-4
-5
-6
-7
-8
26
28
30
32
34
36
38
SNR [dB]
FC=0
1
2
3
4
5
6
7

FIGURE A.3.32

QAM-64 SANS IIS SIR= 22 dB DF= 4

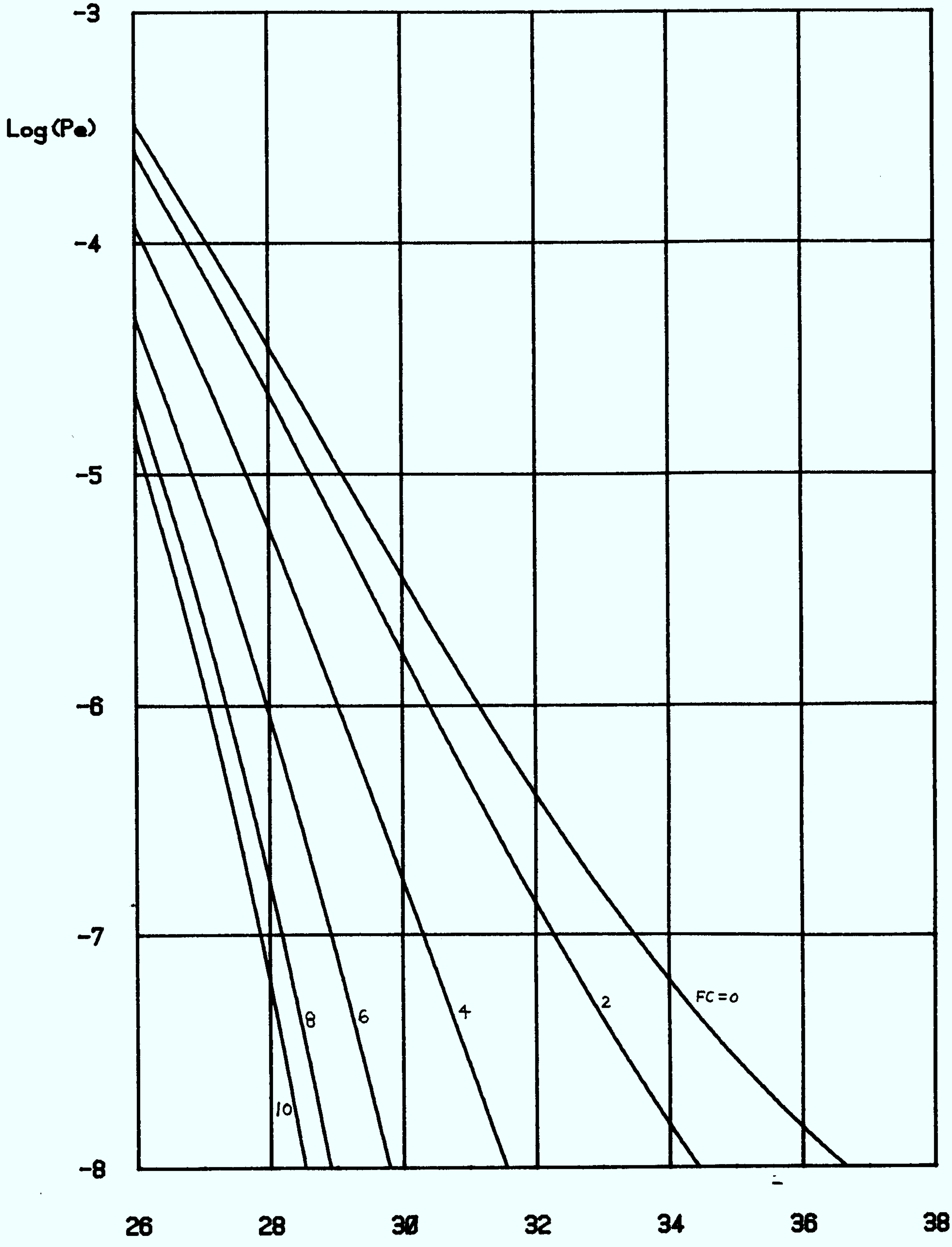


FIGURE A.3.33

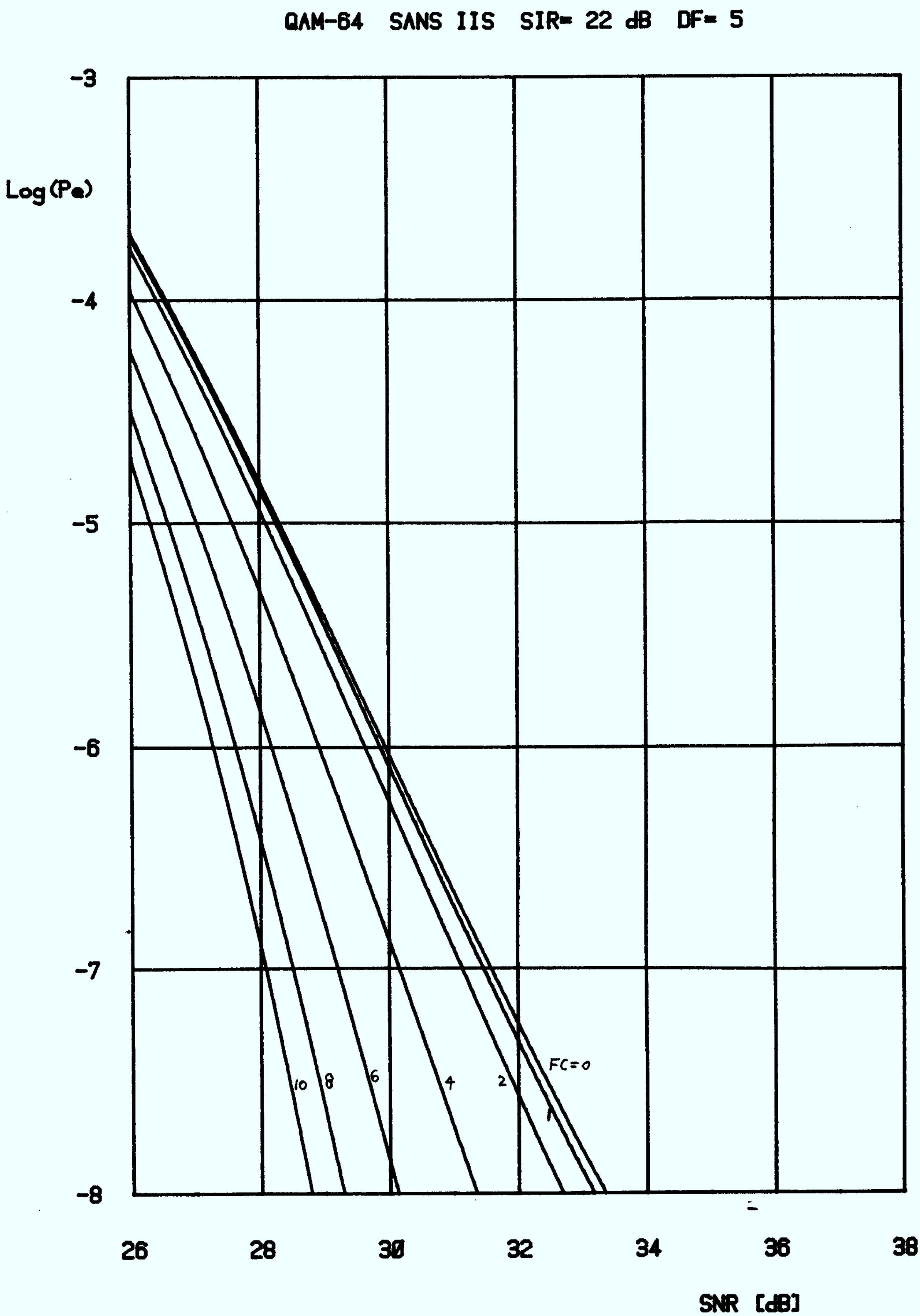

QAM-64 SANS IIS SIR= 22 dB DF= 5
Log(Pe)
-3
-4
-5
-6
-7
-8
26
28
30
32
34
36
38
SNR [dB]
FC=0
1
2
4
6
8
10

FIGURE A.3.34

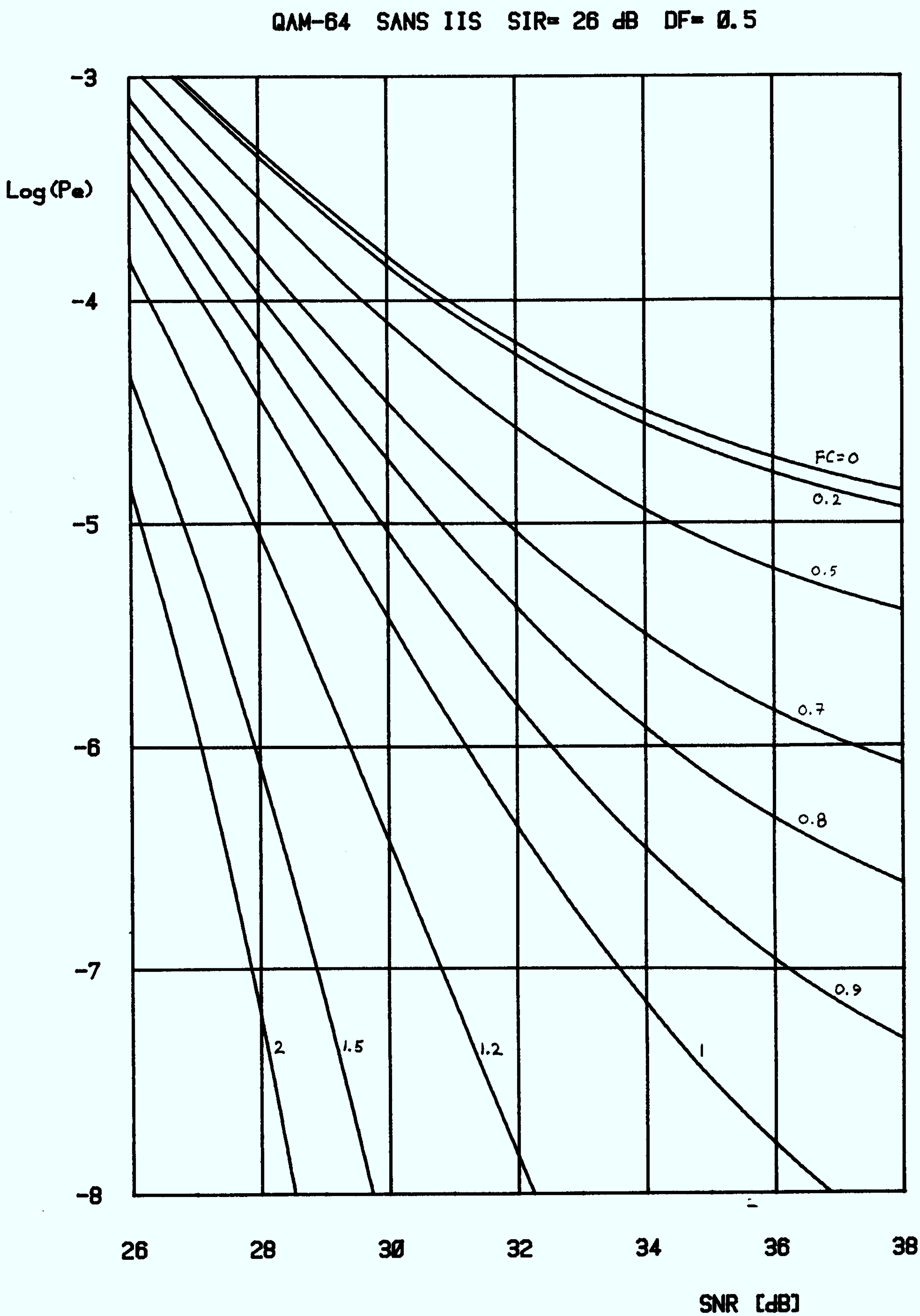

QAM-64 SANS IIS SIR= 26 dB DF= Ø.5
Log(Pe)
-3
-4
-5
-6
-7
-8
FC=0
0.2
0.5
0.7
0.8
0.9
2
1.5
1.2
1
26
28
3Ø
32
34
36
38
SNR [dB]

FIGURE A.3.35

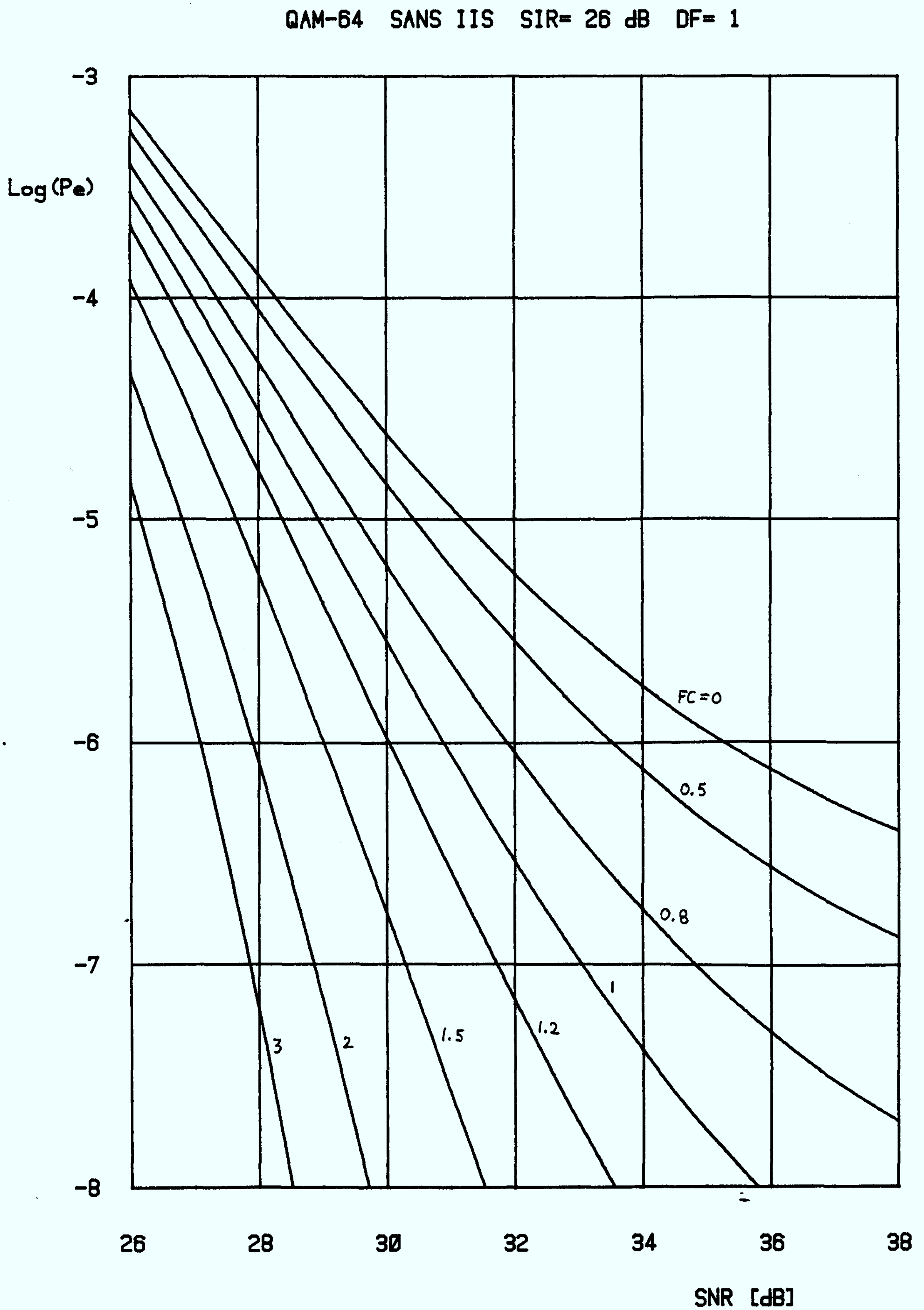

QAM-64 SANS IIS SIR= 26 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
FC=0
0.5
0.8
1
1.2
1.5
2
3
26
28
30
32
34
36
38
SNR [dB]

FIGURE A.3.36

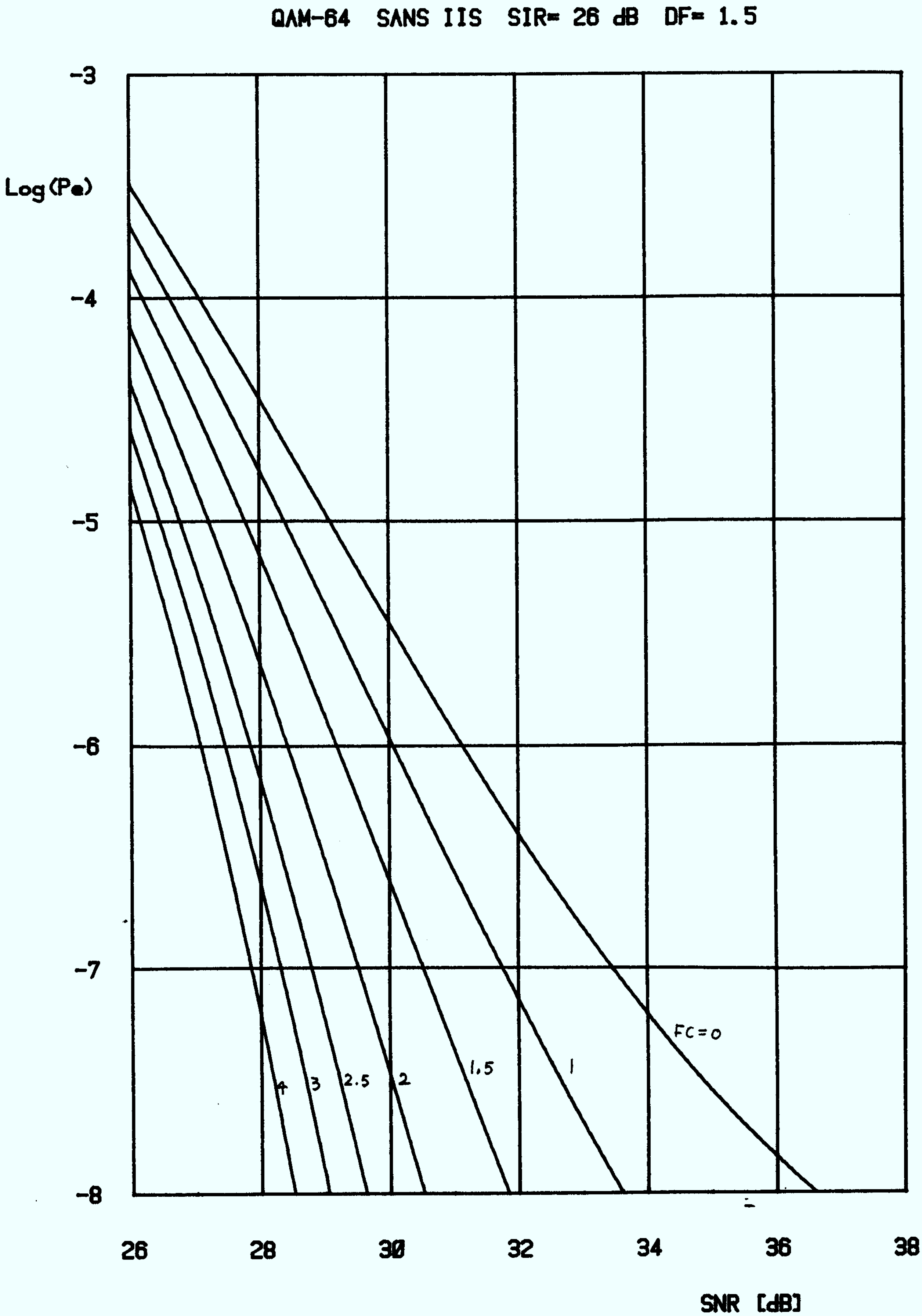

QAM-64 SANS IIS SIR= 26 dB DF= 1.5
Log(Pe)
-3
-4
-5
-6
-7
-8
26
28
30
32
34
36
38
SNR [dB]
FC=0
1
1.5
2
2.5
3
4

FIGURE A.3.37

QAM-64 SANS IIS SIR= 26 dB DF= 2

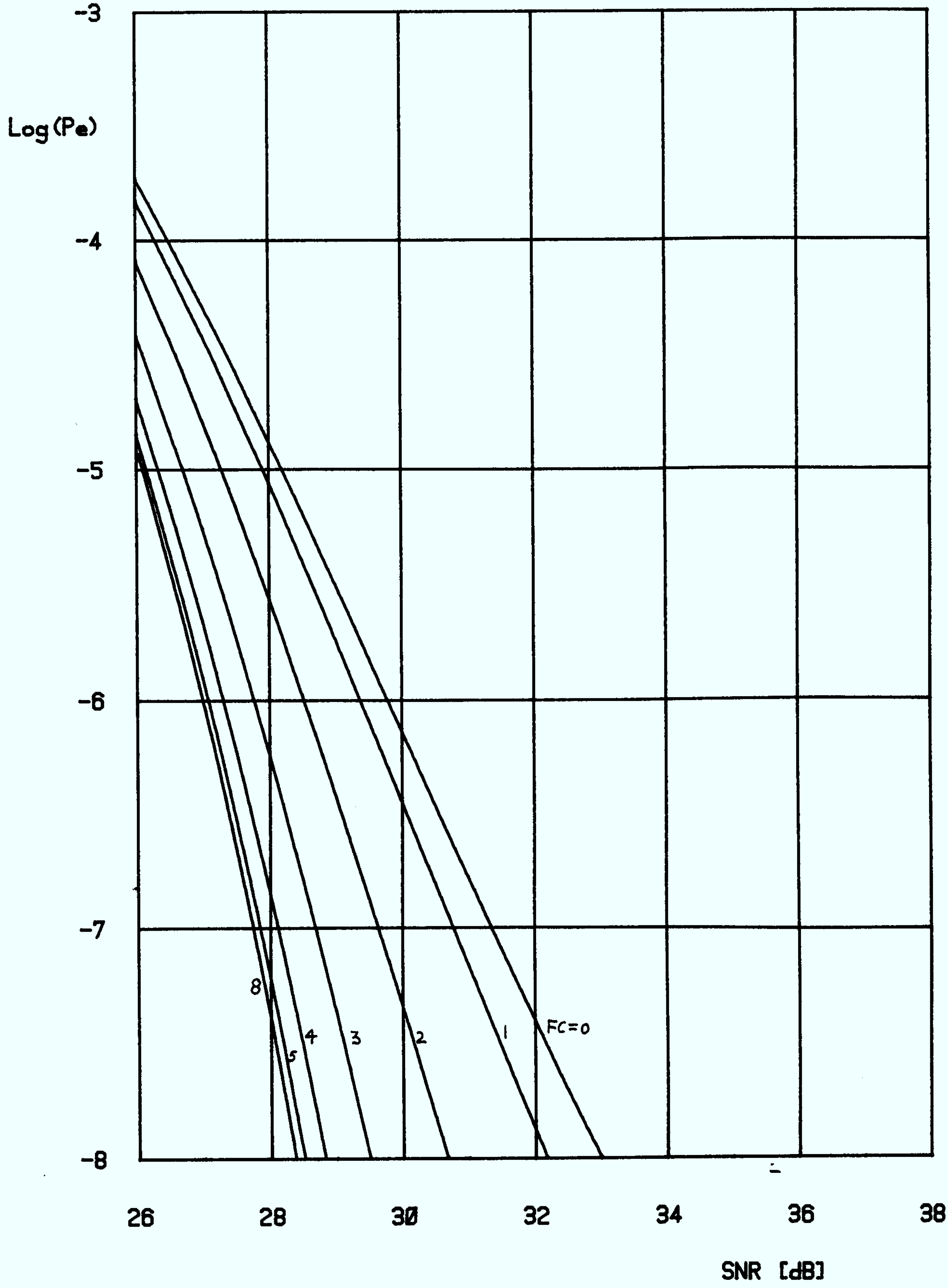


FIGURE A.3.38

QAM-64 SANS IIS SIR= 26 dB DF= 3

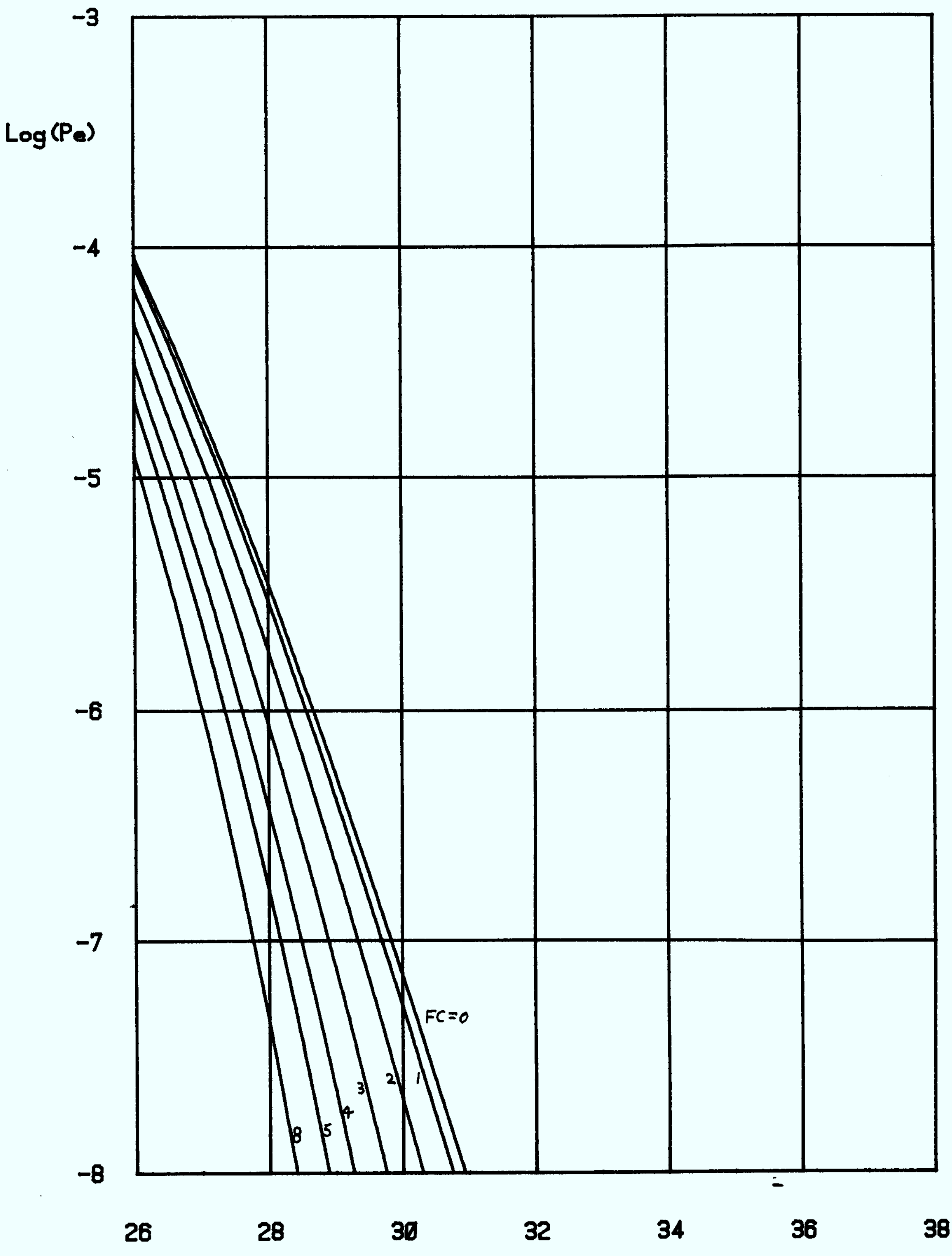


FIGURE A.3.39

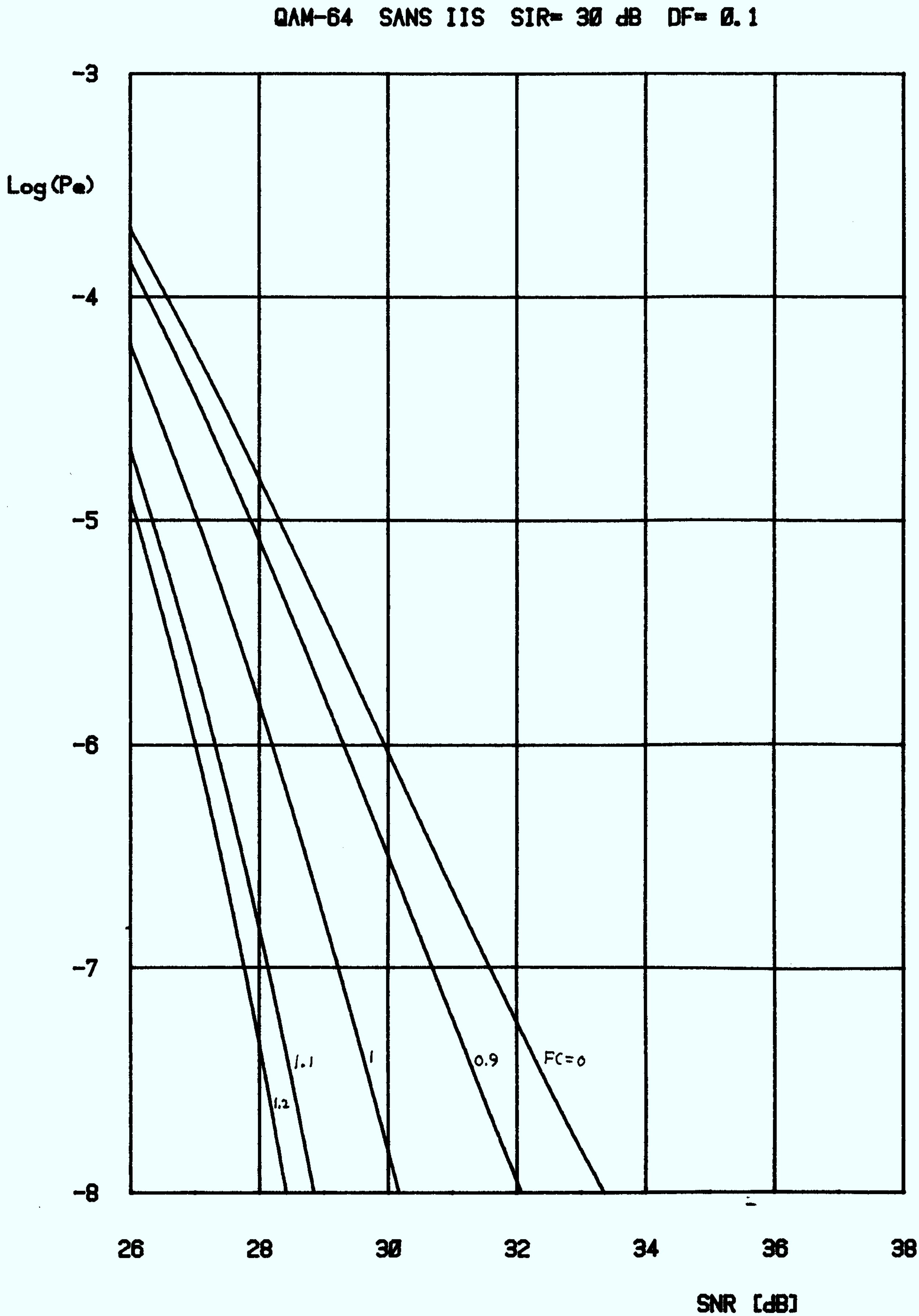

QAM-64 SANS IIS SIR= 30 dB DF= 0.1
Log(Pe)
-3
-4
-5
-6
-7
-8
26
28
30
32
34
36
38
SNR [dB]
1.2
1.1
1
0.9
FC=0

FIGURE A.3.40

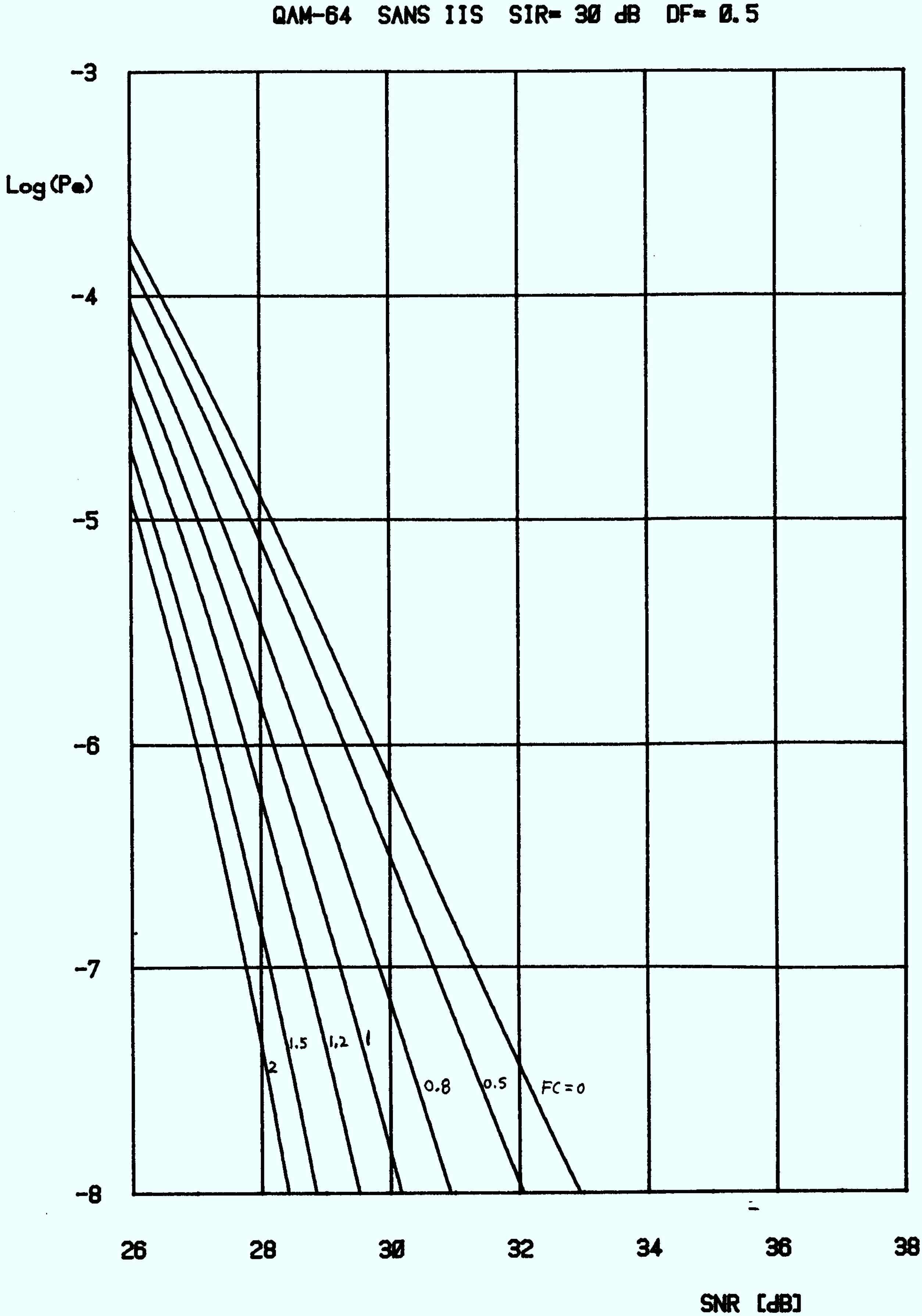

QAM-64 SANS IIS SIR= 30 dB DF= 0.5
-3
Log (Pe)
-4
-5
-6
-7
-8
26
28
30
32
34
36
38
SNR [dB]
2
1.5
1.2
1
0.8
0.5
FC=0

FIGURE A.3.41

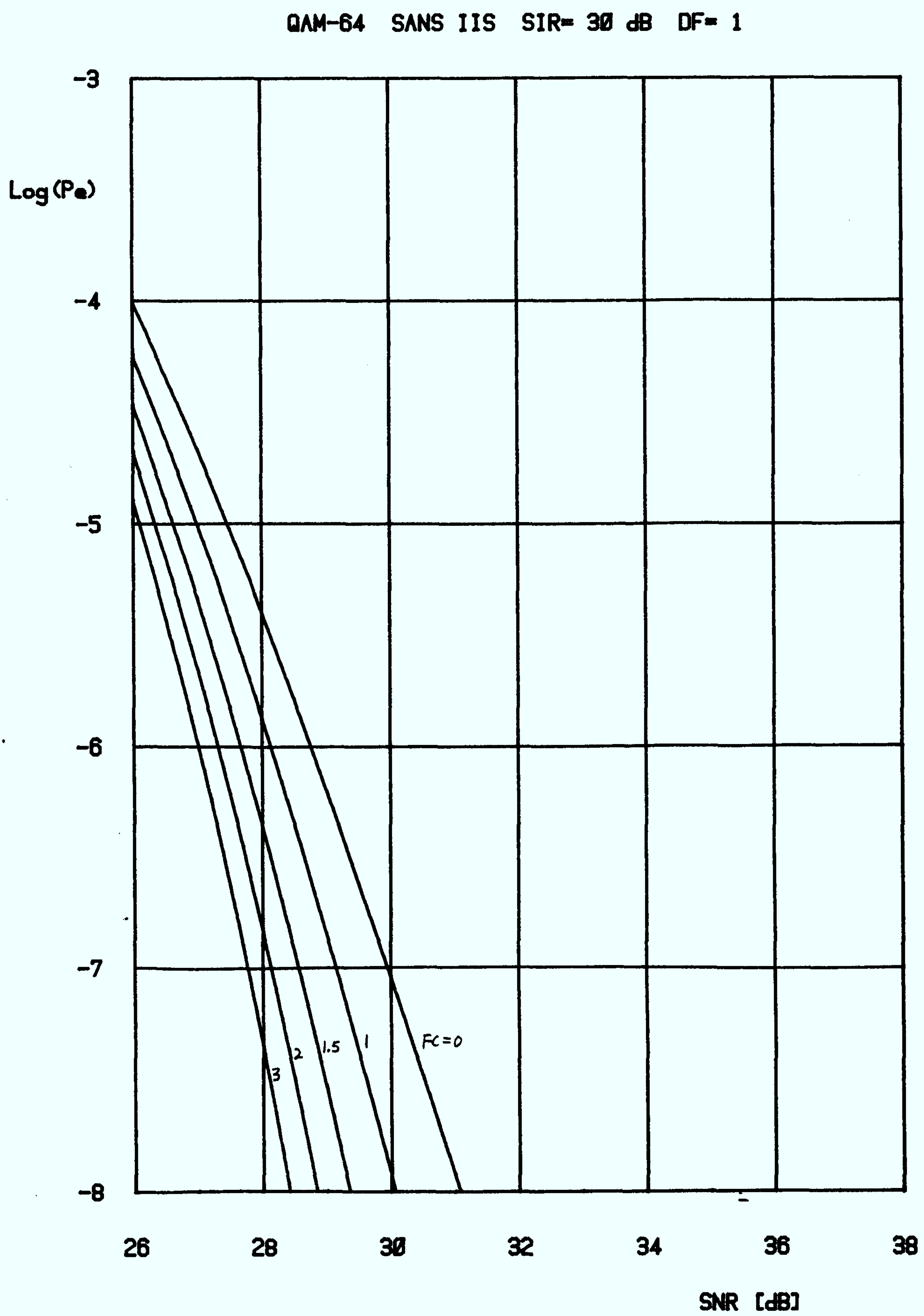

QAM-64 SANS IIS SIR= 30 dB DF= 1
Log(Pe)
-3
-4
-5
-6
-7
-8
26
28
30
32
34
36
38
SNR [dB]
3
2
1.5
1
FC=0

FIGURE A.3.42

ANNEXE 4

LISTING DES PROGRAMMES DÉVELOPPÉS

PROGRAMME QAM

Calcule la probabilité d'erreur moyenne à la sortie d'un récepteur QAM avec ou sans interférences inter-symboles, par la méthode des moments. La réponse du canal est du type Butterworth d'ordre 5 à phase nulle avec une "pulse" de transmission de forme rectangulaire. Le calcul des moments IIS est réalisé par la méthode directe modifiée.

Lignes

1-18 Initialisation des constantes et paramètres et déclarations des variables.

Z : détermine l'ordre du filtre Butterworth modélisant la réponse du canal. La convergence des moments IIS est assurée pour Z = 5.

FIN : détermine la longueur du vecteur des moments à être utilisés par la fonction HERMITE.

KT : constante de pondération pour le calcul de la puissance du bruit à la sortie du filtre Butterworth. La valeur indiquée est pour Z = 5.

PI2 et LN10: constantes 2π et ℓn 10.

22-34 FREQ: calcul de la fonction à intégrer pour le calcul des échantillons de la réponse du canal. Cette fonction constitue la fonction de transfert globale du système.

38-67 ROM: routine d'intégration numérique par l'algorithme de Romberg. Cette fonction calcule l'intégrale de la fonction FREQ entre les bornes BS et BI pour une précision de 10^{-12} ou 15 ittérations.

71-79 REP: calcule la valeur de la réponse du canal à l'instant K. La variable d'intégration est normalisée à 1/T et on intègre de 0 à l'infini en 12 intervalles successifs.

84-104 HERMITE: calcule le résultat de la série de P_e jusqu'à une précision de 10^{-5}. Cette fonction a comme argument la variable X et le vecteur Z contenant les moments d'ordre 2 jusqu'à 2K (FIN = K).

108-140 MOM16: calcul des moments IIS d'ordre 2 jusqu'à 2K pour un récepteur QAM-16 à partir de 4 échantillons du canal.

144-176 MOM36: calcule les moments IIS d'ordre 2 jusqu'à 2K pour un récepteur QAM-36 à partir de 4 échantillons du canal.

180-204 MOM64: calcule les moments IIS d'ordre 2 jusqu'à 2K pour un récepteur QAM-64 à partir de 2 échantillons du canal.

208-254 MOM-4: calcule les moments IIS d'ordre 2 jusqu'à 2K pour un récepteur QAM-4 à partir de 4 échantillons du canal.

258-268 COMB: calcule les combinaisons K parmi N.

272-293 UNION: permet de combiner les moments d'ordre 2 jusqu'à 2K de 2 variables aléatoires indépendantes.

297-342 ERFC: calcule de la fonction d'erreur complémentaire pour $|x|$ entre 0 et 12 par la méthode de Gauss d'ordre 20. Cette formule permet une précision d'environ 6 décimales.

345-408 Programme principal:

CNOB: variable permettant d'initialiser la valeur de base de SNR.

BT: valeur de la largeur de bande du canal normalisée à 1/T.

X1: valeur de la réponse du canal à l'instant t_o de décision.

N: détermine le type de QAM désiré. $M = (2N)^2$

RESI: contient le vecteur final des moments IIS d'ordre 2K pour 16 échantillons du canal.

CNO: valeur du rapport signal sur bruit.

SIGMA: valeur du rapport porteuse à bruit pour le cas où on tient compte de l'IIS (argument de HERMITE).

SIGMA2: valeur du rapport porteuse à bruit sans IIS.

PE1: probabilité d'erreur due au bruit gaussien seulement.

PE2: probabilité d'erreur due au bruit gaussien et à l'IIS.

360-392 Ici, on calcule pour chacun des récepteurs étudié (QAM-4-16-36 ou 64) la valeur des moments d'ordre 2k de l'IIS pour 16 échantillons du canal par différentes combinaisons successives avec la fonction UNION. Notons que les échantillons du canal sont normalisés à X1. Le résultat final est dans RESI et sert pour le calcul de P_e dans HERMITE.

```
0680 --      (*$L+*)
0680 --      PROGRAM QAM(OUTPUT);
0694 --      LABEL 5;
0694 --      CONST
0694 --           Z=5;
0694 --         FIN=200;
0694 --         PI2=6.2831853071795862;
0694 --          KT=1.0166407381;
0694 --         LN10=2.302585092994045 6;
0694 --      TYPE
0694 --         VECT=ARRAY(.1..8.) OF REAL;
0694 --         VECTEUR=ARRAY(.1..FIN.) OF REAL;
0694 --          SPEC=ARRAY(.1..2,1..15.) OF REAL;
0694 --      VAR
0694 --         VARIAB:VECT;
06D8 --         VECT1,VECT2,RES1,RES2,RES3,RES4:VECTEUR;
2C58 --         I,J,L,N,K,M,POSI,PAS,P,Q,R,V,S:INTEGER;
2C8C --         CNOB,CNO,SIGMA,SIGMA2,BT,X1,PE1,PE2:REAL;
2CD0 --
2CD0 --      (*-----------------------------------------------------------------------*)
2CD0 --
2CD0 --   A  FUNCTION FREQ(X,BT:REAL;K:INTEGER):REAL;
005C --      (*CALCUL DE LA FONCTION A INTEGRER*)
005C --      VAR Y,S:REAL;
0000 0-   A  BEGIN
0018 --         IF X=0 THEN FREQ:=2
003E 1-               ELSE BEGIN Y:=2*(COS(PI2*X*K))*SIN(PI2*X/2)/(PI2*X/2);
0108 --                      S:=2*Z*LN(2*X/BT);
0166 --                      IF ABS(S)>70 THEN
0190 --                           IF X<(BT/2) THEN FREQ:=Y
01C8 --                                       ELSE FREQ:=0
01DE --                           ELSE FREQ:=Y/SQRT(1+(2*X/BT)**(2*Z));
0276 -1                 END;
0276 -0   A  END;
029C --
029C --      (*-----------------------------------------------------------------------*)
029C --
029C --   A  FUNCTION ROM(BT,BS,BI:REAL;K:INTEGER):REAL;
0064 --      (*CALCUL DE LA REPONSE DU CANAL*)
0064 --      LABEL 5;
0064 --      VAR
0064 --          H,L:REAL;
0078 --          I,J,P:INTEGER;
0084 --          R:SPEC;
0000 0-   A  BEGIN
0018 --          H:=BS-BI;
003A --          R(.1,1.):=H*(FREQ(BI,BT,K)+FREQ(BS,BT,K))/2;
00E6 --          I:=1;
00EE 1-          REPEAT
00EE --          I:=I+1;
0100 --                    L:=0;
0116 --                    FOR P:=1 TO 2**(I-2) DO
0154 2-                        BEGIN
0154 --                        L:=L+FREQ(BI+((2*P-1)*H/2),BT,K);
01F0 -2                        END;
0212 --                    R(.2,1.):=(R(.1,1.)+H*L)/2;
025A --                    FOR J:=2 TO I DO
0286 2-                        BEGIN
```

```
0286 --                     R(.2,J.):=R(.2,J-1.)*(4**(J-1))-R(.1,J-1.);
030E --                     R(.2,J.):=R(.2,J.)/(4**(J-1)-1);
0370 -2                     END;
0392 --                HI=H/2;
03BA --                FOR J:=1 TO I DO R(.1,J.):=R(.2,J.);
043E --           IF R(.2,I.)=0 THEN GOTO 5;
047A -1           UNTIL (ABS((R(.2,I.)-R(.2,I-1.))/R(.2,I.))<1E-12) OR (I=15);
0518 --           5:ROM:=R(.2,I.);
053C -0 A    END;
056C --
056C --      (*------------------------------------------------------------------*)
056C --
056C -- A      FUNCTION REP(BT:REAL;K:INTEGER):REAL;
0054 --      VAR
0054 --           I:INTEGER;
0058 --           S:REAL;
0000 0- A    BEGIN
0018 --           S:=0;
002E --         FOR I:=0 TO 12 DO S:=S+ROM(BT,I+1,I,K);
00E2 --         REP:=S;
00F4 -0 A    END;
0114 --
0114 --
0114 --      (*-----------------------------------------------------------------*)
0114 --
0114 -- A    FUNCTION HERMITE(X:REAL;Z:VECTEUR):REAL;
0690 --      VAR
0690 --          J:INTEGER;
0694 --          ARG,TRANS,A,B:REAL;
06B8 --          V:VECTEUR;
0000 0- A    BEGIN
0018 --          ARG:=(1/SQRT(PI2/2))*EXP(-X*X);
0082 --          V(.1.):=X**3;
00A2 --          V(.2.):=(X**4)*(8*(X**3)-12*X)/24;
0132 --          J:=2;
013A --          TRANS:=V(.1.)*Z(.1.)+V(.2.)*Z(.2.);
017C 1-          REPEAT
017C --               J:=J+1;
018E --               A:=(1-((4*J-5)/(2*X*X)))*V(.J-1.);
0232 --               B:=V(.J-2.)*(J-2)/(J-1);
02A0 --               V(.J.):=(2*(X**4)/(J*(2*J-1)))*(A-B);
0344 --               TRANS:=TRANS+V(.J.)*Z(.J.);
039A -1          UNTIL (ABS(V(.J.)*Z(.J.)/TRANS)<1E-5) OR (J=FIN);
041C --          HERMITE:=TRANS*ARG;
043E --          WRITE('      ',J:3);
046E -0 A    END;
04B4 --
04B4 --      (*-----------------------------------------------------------------*)
04B4 --
04B4 -- A    PROCEDURE MOM16(VAR V:VECTEUR;X:VECT);
0088 --      (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --      VAR
0088 --         Z:ARRAY(.0..255.) OF REAL;
0888 --         I,J,K,L:INTEGER;
0000 0- A    BEGIN
0012 --         FOR I:=0 TO 3 DO
002E 1-         BEGIN
002E --              FOR J:=0 TO 3 DO
```

```
004A 2-              BEGIN
004A --                   FOR K:=0 TO 3 DO
0066 3-                   BEGIN
0066 --                        FOR L:=0 TO 3 DO
0082 4-                        BEGIN
0082 --                          Z(.I*64+J*16+K*4+L.):=SQR(((2*I-3)*X(.1.)+(2*J-3)*X(.2.)
00FC --                          +(2*K-3)*X(.3.)+(2*L-3)*X(.4.)));
0188 -4                        END;
01AA -3                   END;
01CC -2              END;
01EE -1         END;
0210 --         V(.1.):=0;
022A --         FOR I:=0 TO 255 DO V(.1.):=V(.1.)+Z(.I.);
02A4 --         FOR J:=2 TO FIN DO
02C2 1-         BEGIN
02C2 --                   V(.J.):=0;
02F2 --                   FOR I:=0 TO 255 DO
030E 2-                   BEGIN
030E --                   IF (LN(Z(.I.))/LN10)*J > -70 THEN
036A --                        V(.J.):=V(.J.)+Z(.I.)**J;
03E0 -2                   END;
0402 -1         END;
0424 --         FOR I:=1 TO FIN DO V(.I.):=V(.I.)/256;
04C0 -0 A  END;
04F4 --
04F4 --    (*-----------------------------------------------------------------------*)
04F4 --
04F4 -- A  PROCEDURE MOM36(VAR V:VECTEUR;X:VECT);
0088 --    (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --    VAR
0088 --         Z:ARRAY(.0..1295.) OF REAL;
2908 --         I,J,K,L:INTEGER;
0000 0- A  BEGIN
0012 --         FOR I:=0 TO 5 DO
003A 1-         BEGIN
003A --              FOR J:=0 TO 5 DO
0062 2-              BEGIN
0062 --                   FOR K:=0 TO 5 DO
008A 3-                   BEGIN
008A --                        FOR L:=0 TO 5 DO
0082 4-                        BEGIN
0082 --                          Z(.I*216+J*36+K*6+L.):=SQR(((2*I-5)*X(.1.)+(2*J-5)*X(.2.)
0150 --                          +(2*K-5)*X(.3.)+(2*L-5)*X(.4.)));
01E8 -4                        END;
021C -3                   END;
0250 -2              END;
0284 -1         END;
0288 --         V(.1.):=0;
02D2 --         FOR I:=0 TO 1295 DO V(.1.):=V(.1.)+Z(.I.);
0370 --         FOR J:=2 TO FIN DO
039A 1-         BEGIN
039A --                   V(.J.):=0;
03D0 --                   FOR I:=0 TO 1295 DO
03F8 2-                   BEGIN
03F8 --                   IF (LN(Z(.I.))/LN10)*J > -70 THEN
0460 --                        V(.J.):=V(.J.)+Z(.I.)**J;
04EE -2                   END;
0522 -1         END;
```

```
0556 --        FOR I:=1 TO FIN DO V(.I.):=V(.I.)/1296;
061C -0 A   END;
0658 --
0658 --     (*-----------------------------------------------------------------------*)
0658 --
0658 -- A   PROCEDURE MOM64(VAR V:VECTEUR;X1,X2:REAL);
0058 --     (*POUR CALCULER LES MOMENTS NORMALISES A PARTIR
0058 --       DE DEUX ECHANTILLONS. *)
0058 --     VAR
0058 --        I,J:INTEGER;
0060 --        Z:ARRAY(.0..63.) OF REAL;
0000 0- A   BEGIN
0012 --        FOR I:=0 TO 7 DO
002E 1-        BEGIN
002E --            FOR J:=0 TO 7 DO
004A 2-            BEGIN
004A --                Z(.8*I+J.):=SQR(((2*I-7)*X1+(2*J-7)*X2));
00D0 -2            END;
00F2 -1        END;
0114 --        FOR J:=1 TO FIN DO
0132 1-        BEGIN
0132 --            V(.J.):=0;
0162 --            FOR I:=0 TO 63 DO
017E 2-            BEGIN
017E --                IF (LN(Z(.I.))/LN10)*J > -70 THEN
01DA --                    V(.J.):=V(.J.)+Z(.I.)**J;
0250 -2            END;
0272 --            V(.J.):=V(.J.)/64;
02CE -1        END;
02F0 -0 A   END;
0318 --
0318 --     (*-----------------------------------------------------------------------*)
0318 --
0318 -- A   PROCEDURE MOM4(VAR V:VECTEUR;X:VECT);
0088 --     (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --     VAR
0088 --        Z:ARRAY(.0..255.) OF REAL;
0888 --        I,J,K,L,M,N,P,Q:INTEGER;
0000 0- A   BEGIN
0012 --      FOR I:=0 TO 1 DO
002E 1-      BEGIN
002E --       FOR J:=0 TO 1 DO
004A 2-       BEGIN
004A --        FOR K:=0 TO 1 DO
0066 3-        BEGIN
0066 --         FOR L:=0 TO 1 DO
0082 4-         BEGIN
0082 --         FOR M:=0 TO 1 DO
009E 5-         BEGIN
009E --          FOR N:=0 TO 1 DO
00BA 6-          BEGIN
00BA --           FOR P:=0 TO 1 DO
00D6 7-           BEGIN
00D6 --            FOR Q:=0 TO 1 DO
00F2 8-            BEGIN
00F2 --                Z(.I*128+J*64+K*32+L*16+M*8+N*4+P*2+Q.):=
0164 --                SQR(((2*I-1)*X(.1.)+(2*J-1)*X(.2.)
01A4 --                +(2*K-1)*X(.3.)+(2*L-1)*X(.4.)+(2*M-1)*X(.5.)
```

```
023A --                +(2*N-1)*X(.6.)+(2*P-1)*X(.7.)+(2*Q-1)*X(.8.)));
02F8 -8             END;
031A -7            END;
033C -6           END;
035E -5          END;
0380 -4         END;
03A2 -3        END;
03C4 -2       END;
03E6 -1      END;
0408 --        V(.1.):=0;
0422 --        FOR I:=0 TO 255 DO V(.1.):=V(.1.)+Z(.I.);
049C --        FOR J:=2 TO FIN DO
04BA 1-        BEGIN
04BA --                V(.J.):=0;
04EA --                FOR I:=0 TO 255 DO
0506 2-                BEGIN
0506 --                IF (LN(Z(.I.))/LN10)*J > -70 THEN
0562 --                    V(.J.):=V(.J.)+Z(.I.)**J;
05D8 -2                END;
05FA -1        END;
061C --        FOR I:=1 TO FIN DO V(.I.):=V(.I.)/256;
06B8 -0 A    END;
06F8 --
06F8 --      (*-------------------------------------------------------------------*)
06F8 --
06F8 -- A    FUNCTION COMB(X:REAL;K,N:INTEGER):REAL;
0058 --      (*POUR CALCULER LES COMBINAISONS "K PARMI N" *)
0058 --      VAR
0058 --          I:INTEGER;
0000 0- A    BEGIN
0018 --          FOR I:=N DOWNTO K+1 DO
004C 1-               BEGIN
004C --                    X:=X*I/(I-K);
0096 -1               END;
00B8 --          COMB:=X;
00CA -0 A    END;
00E8 --
00E8 --      (*-------------------------------------------------------------------*)
00E8 --
00E8 -- A    PROCEDURE UNION(VAR X,Y,Z:VECTEUR);
004C --      (*POUR COMBINER DEUX MOMENTS,*)
004C --      VAR
004C --         I,J:INTEGER;
0054 --         B,C:REAL;
0000 0- A    BEGIN
0012 --         FOR I:=1 TO FIN DO Z(.I.):=X(.I.)+Y(.I.);
00C2 --         FOR I:=2 TO FIN DO
00E0 1-             BEGIN
00E0 --                 J:=1;
00E8 2-                 REPEAT
00E8 --                 IF (X(.J.)<>0) AND (Y(.I-J.)<>0) THEN
0188 3-                 BEGIN
0188 --                     B:=LN(Y(.I-J.));
01C8 --                     C:=LN(COMB(X(.J.),2*J,2*I));
0230 --                     IF (B+C)/LN10>-70 THEN
026C --                     Z(.I.):=Z(.I.)+EXP(B+C);
02DC -3                 END;
02DC --                 J:=J+1
```

```
02E6 -2                 UNTIL J>=I!
0300 -1             END!
0322 -0 A      END!
034C --
034C --        (*-----------------------------------------------------------------------*)
034C --
034C -- A      FUNCTION ERFC(VAR X!REAL)!REAL!
004C --        (*CALCUL DE LA FONCTION D'ERREUR COMPLEMENTAIRE PAR LA METHODE
004C --        DE GAUSS D'ORDRE 20*)
004C --
004C --        TYPE
004C --             GAUSS=ARRAY(.1..10.) OF REAL!
004C --        VAR
004C --             T,W!GAUSS!
00F0 --             A,B,R,C!REAL!
0110 --             I!INTEGER!
0114 --
0114 -- B      FUNCTION F(VAR X!REAL)!REAL!
004C --        CONST PI=3,141592653589791
0000 0- B      BEGIN
0018 --             F!=(2/SQRT(PI))*EXP(-X*X)!
0072 -0 B      END!
0094 --
0000 0- A      BEGIN
0018 --             T(.1.)!=0,0765265211334971W(.1.)!=0,152753387130725!
0028 --             T(.2.)!=0,227785851141645!W(.2.)!=0,149172986472603!
0038 --             T(.3.)!=0,373706088715419!W(.3.)!=0,142096109318382!
0048 --             T(.4.)!=0,510867001950827!W(.4.)!=0,131688638449176!
0058 --             T(.5.)!=0,636053680726515!W(.5.)!=0,118194531961518!
0068 --             T(.6.)!=0,746331906460150!W(.6.)!=0,101930119817240!
0078 --             T(.7.)!=0,839116971822218!W(.7.)!=0,083276741576704!
0088 --             T(.8.)!=0,912234428251325!W(.8.)!=0,062672048334109!
0098 --             T(.9.)!=0,963971927277913!W(.9.)!=0,040601429800386!
00A8 --             T(.10.)!=0,993128599185094!W(.10.)!=0,017614007139152!
00B8 --             IF ABS(X)>12 THEN R!=0 ELSE
0100 1-                   BEGIN
0100 --                   A!=(13-ABS(X))/2!
0144 --                   B!=(13+ABS(X))/2!
0188 --                   R!=0!
019E --                   FOR I!=1 TO 10 DO
01BC 2-                         BEGIN
01BC --                         C!=B+A*T(.I.)!
0200 --                         R!=R+W(.I.)*F(C)!
0252 -2                         END!
0274 --                   FOR I!=1 TO 10 DO
0292 2-                         BEGIN
0292 --                         C!=B-A*T(.I.)!
02D6 --                         R!=R+W(.I.)*F(C)!
0328 -2                         END!
034A -1                   END!
034A --             IF X<0 THEN ERFC!=2-A*R ELSE ERFC!=A*R!
03D2 -0 A      END!
0494 --
0494 --
0000 0-        BEGIN
003A --           CNOB!=18!
0058 --              WRITELN('                    FILTRE BUTTERWORTH ORDRE 5')!
0076 --             WRITELN!
```

```
0082 --        WRITELN;WRITELN;
009A --        FOR Q:=0 TO 0 DO
00C2 1-        BEGIN
00C2 --        BT:=1,05+0,03*Q;
00EE --        WRITE('         BT=',BT:5:2);
012C --     X1:=ABS(REP(BT,0));
015E --          FOR N:=2 TO 2 DO
0188 2-          BEGIN
0188 --          WRITELN('          MODULATION QAM=',SQR(2*N):3);
01D0 --     WRITELN;WRITELN;
01E8 3-        CASE N OF
01F6 4-              1: BEGIN
01F6 --                 FOR I:=1 TO 8 DO VARIAB(,I,):=REP(BT,I)/X1;
02B4 --                  MOM4(VECT1,VARIAB);
02C8 --                 UNION(VECT1,VECT1,RES1);
02EA -4                 END;
02EE 4-              2: BEGIN
02EE --                 FOR I:=1 TO 4 DO VARIAB(,I,):=REP(BT,2*I)/X1;
03B4 --                 MOM16(RES1,VARIAB);
03C8 --                 FOR I:=1 TO 4 DO VARIAB(,I,):=REP(BT,2*I-1)/X1;
0492 --                 MOM16(RES2,VARIAB);
04A6 --                 UNION(RES1,RES2,VECT1);
04C8 --                 UNION(VECT1,VECT1,RES1);
04EA -4                 END;
04EE 4-              3: BEGIN
04EE --                FOR I:=1 TO 4 DO VARIAB(,I,):=REP(BT,2*I)/X1;
05B4 --                 MOM36(RES1,VARIAB);
05C8 --                 FOR I:=1 TO 4 DO VARIAB(,I,):=REP(BT,2*I-1)/X1;
0692 --                 MOM36(RES2,VARIAB);
06A6 --                 UNION(RES1,RES2,VECT1);
06C8 --                 UNION(VECT1,VECT1,RES1);
06EA -4                 END;
06EE 4-              4: BEGIN
06EE --                 FOR I:=1 TO 8 DO VARIAB(,I,):=REP(BT,I)/X1;
07AC --                 MOM64(RES1,VARIAB(,1,),VARIAB(,8,));
07DE --                 MOM64(RES2,VARIAB(,2,),VARIAB(,7,));
0810 --                 UNION(RES1,RES2,VECT1);
0832 --                 MOM64(RES1,VARIAB(,3,),VARIAB(,6,));
0864 --                 MOM64(RES2,VARIAB(,4,),VARIAB(,5,));
0896 --                 UNION(RES1,RES2,VECT2);
08B8 --                 UNION(VECT1,VECT2,RES3);
08D8 --                 UNION(RES3,RES3,RES1);
08F6 -4                 END
08F6 -3              END;
0922 --        FOR J:=0TO 12 DO
094A 3-        BEGIN
094A --              CNO:=J+CNOB;
0984 --              WRITE('    SNR=',CNO:5:1,' DB');
09D4 --              SIGMA:=X1*SQRT(3*(10**(CNO/10))/(2*(SQR(2*N)-1)*BT*KT));
0AB6 --              SIGMA2:=(SIGMA/X1)*SQRT(BT*KT);
0B0E --              PE1:=(2-1/N)*0,5*ERFC(SIGMA2);
0B88 --              WRITE('       PE1=',PE1:9);
0BC4 --              PE2:=(2-1/N)*(0,5*ERFC(SIGMA)+HERMITE(SIGMA,RES1));
0C86 --              WRITELN('    TERMES          PE2=',PE2:9);
0CC8 -3          END;
0CFC --        WRITELN;
0D08 -2       END;
0D3C --          WRITELN;
```

```
0D48 =1     END;
0D7C =0     5; END.
```

PROGRAMME QAMBS

En présence d'une interférence sinusoïdale d'amplitude constante et de phase également répartie, calcule la probabilité d'erreur moyenne à la sortie d'un récepteur QAM avec ou sans IIS par la méthode des moments. La réponse du canal est du type Butterworth d'ordre 5 et à phase nulle, avec une "pulse" à la transmission de forme rectangulaire. Le calcul des moments IIS est réalisé par la méthode directe modifiée.

Lignes

1-346 Voir QAM

348-446 Programme principal

SIR: rapport signal à interférence.

PE1: probabilité d'erreur due au bruit gaussien seulement.

PE2: probabilité d'erreur due au bruit gaussien avec IIS.

PE3: probabilité d'erreur due au bruit gaussien avec IIS et interférence sinusoïdale.

421-426 On calcule les moments de l'interférence sinusoïdale. Le résultat dans RES2 est combiné avec les moments IIS dans RES1 par la procédure UNION. Le résultat final est dans VECT1.

```
0680 --      (*$L+*)
0680 --      PROGRAM QAMBS(OUTPUT);
0694 --      LABEL 5;
0694 --      CONST
0694 --           Z=5;
0694 --         FIN=200;
0694 --         PI2=6.2831853071795862;
0694 --          KT=1.0166407738;
0694 --         LN10=2.302585092994056;
0694 --      TYPE
0694 --         VECT=ARRAY(.1..8.) OF REAL;
0694 --         VECTEUR=ARRAY(.1..FIN.) OF REAL;
0694 --          SPEC=ARRAY(.1..2,1..15.) OF REAL;
0694 --      VAR
0694 --         VARIAB:VECT;
06D8 --         VECT1,VECT2,RES1,RES2,RES3,RES4:VECTEUR;
2C58 --         I,J,L,N,K,M,POSI,PAS,P,Q,R,V,S:INTEGER;
2C8C --         CNOB,CNO,SIGMA,SIGMA2,BT,X1,SIR:REAL;
2CC8 --          PE1,PE2,PE3:REAL;
2CE0 --
2CE0 --      (*--------------------------------------------------------------------*)
2CE0 --
2CE0 --   A  FUNCTION FREQ(X,BT:REAL;K:INTEGER):REAL;
005C --      (*CALCUL DE LA FONCTION A INTEGRER*)
005C --      VAR Y,S:REAL;
0000 0-   A  BEGIN
0018 --         IF X=0 THEN FREQ:=2
003E 1-            ELSE BEGIN Y:=2*(COS(PI2*X*K))*SIN(PI2*X/2)/(PI2*X/2);
0108 --                    S:=2*Z*LN(2*X/BT);
0166 --                    IF ABS(S)>70 THEN
0190 --                       IF X<(BT/2) THEN FREQ:=Y
01C8 --                           ELSE FREQ:=0
01DE --                       ELSE FREQ:=Y/SQRT(1+(2*X/BT)**(2*Z));
0276 -1                 END;
0276 -0   A  END;
029C --
029C --      (*--------------------------------------------------------------------*)
029C --
029C --   A  FUNCTION ROM(BT,BS,BI:REAL;K:INTEGER):REAL;
0064 --      (*CALCUL DE LA REPONSE DU CANAL*)
0064 --      LABEL 5;
0064 --      VAR
0064 --          H,L:REAL;
0078 --          I,J,P:INTEGER;
0084 --          R:SPEC;
0000 0-   A  BEGIN
0018 --          H:=BS-BI;
003A --          R(.1,1.):=H*(FREQ(BI,BT,K)+FREQ(BS,BT,K))/2;
00E6 --          I:=1;
00EE 1-          REPEAT
00EE --          I:=I+1;
0100 --                  L:=0;
0116 --                  FOR P:=1 TO 2**(I-2) DO
0154 2-                      BEGIN
0154 --                      L:=L+FREQ(BI+((2*P-1)*H/2),BT,K);
01F0 -2                      END;
0212 --                  R(.2,1.):=(R(.1,1.)+H*L)/2;
025A --                  FOR J:=2 TO I DO
```

```
0286 2-                              BEGIN
0286 --                              R(.2,J.):=R(.2,J-1.)*(4**(J-1))-R(.1,J-1.);
030E --                              R(.2,J.):=R(.2,J.)/(4**(J-1)-1);
0370 -2                              END;
0392 --                       H:=H/2;
038A --                       FOR J:=1 TO I DO R(.1,J.):=R(.2,J.);
043E --             IF R(.2,I.)=0 THEN GOTO 5;
047A -1             UNTIL (ABS((R(.2,I.)-R(.2,I-1.))/R(.2,I.))<1E-12) OR (I=15);
0518 --             5:ROM:=R(.2,I.);
053C -0 A      END;
056C --
056C --        (*--------------------------------------------------------------------*)
056C --
056C -- A        FUNCTION REP(BT:REAL;K:INTEGER):REAL;
0054 --        VAR
0054 --             I:INTEGER;
0058 --             S:REAL;
0000 0- A      BEGIN
0018 --             S:=0;
002E --           FOR I:=0 TO 12 DO S:=S+ROM(BT,I+1,I,K);
00E2 --           REP:=S;
00F4 -0 A      END;
0114 --
0114 --
0114 --        (*--------------------------------------------------------------------*)
0114 --
0114 --
0114 -- A      FUNCTION HERMITE(X:REAL;Z:VECTEUR):REAL;
0690 --        VAR
0690 --             J:INTEGER;
0694 --             ARG,TRANS,A,B:REAL;
06B8 --             V:VECTEUR;
0000 0- A      BEGIN
0018 --             ARG:=(1/SQRT(PI2/2))*EXP(-X*X);
0082 --             V(.1.):=X**3;
00A2 --             V(.2.):=(X**4)*(8*(X**3)-12*X)/24;
0132 --             J:=2;
013A --             TRANS:=V(.1.)*Z(.1.)+V(.2.)*Z(.2.);
017C 1-             REPEAT
017C --                  J:=J+1;
018E --                  A:=(1-((4*J-5)/(2*X*X)))*V(.J-1.);
0232 --                  B:=V(.J-2.)*(J-2)/(J-1);
02A0 --                  V(.J.):=(2*(X**4)/(J*(2*J-1)))*(A-B);
0344 --                  TRANS:=TRANS+V(.J.)*Z(.J.);
039A -1             UNTIL (ABS(V(.J.)*Z(.J.)/TRANS)<1E-5) OR (J=FIN);
041C --             HERMITE:=TRANS*ARG;
043E --             WRITE('      ',J:3);
046E -0 A      END;
04B4 --
04B4 --        (*--------------------------------------------------------------------*)
04B4 --
04B4 -- A      PROCEDURE MOM16(VAR V:VECTEUR;X:VECT);
0088 --        (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS,*)
0088 --        VAR
0088 --           Z:ARRAY(.0..255.) OF REAL;
0888 --           I,J,K,L:INTEGER;
0000 0- A      BEGIN
0012 --           FOR I:=0 TO 3 DO
```

```
002E 1-          BEGIN
002E --              FOR J:=0 TO 3 DO
004A 2-              BEGIN
004A --                  FOR K:=0 TO 3 DO
0066 3-                  BEGIN
0066 --                      FOR L:=0 TO 3 DO
0082 4-                      BEGIN
0082 --                        Z(.I*64+J*16+K*4+L.):=SQR(((2*I-3)*X(.1.)+(2*J-3)*X(.2.)
00FC --                        +(2*K-3)*X(.3.)+(2*L-3)*X(.4.)));
0188 -4                      END;
01AA -3                  END;
01CC -2             END;
01EE -1          END;
0210 --          V(.1.):=0;
022A --          FOR I:=0 TO 255 DO V(.1.):=V(.1.)+Z(.I.);
02A4 --          FOR J:=2 TO FIN DO
02C2 1-          BEGIN
02C2 --                  V(.J.):=0;
02F2 --                  FOR I:=0 TO 255 DO
030E 2-                  BEGIN
030E --                  IF (LN(Z(.I.))/LN10)*J > -70 THEN
036A --                      V(.J.):=V(.J.)+Z(.I.)**J;
03E0 -2                  END;
0402 -1          END;
0424 --          FOR I:=1 TO FIN DO V(.I.):=V(.I.)/256;
04C0 -0 A    END;
04F4 --
04F4 --      (*--------------------------------------------------------------------------*)
04F4 --
04F4 -- A    PROCEDURE MOM36(VAR V:VECTEUR;X:VECT);
0088 --      (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --      VAR
0088 --          Z:ARRAY(.0..1295.) OF REAL;
2908 --          I,J,K,L:INTEGER;
0000 0- A    BEGIN
0012 --          FOR I:=0 TO 5 DO
003A 1-          BEGIN
003A --              FOR J:=0 TO 5 DO
0062 2-              BEGIN
0062 --                  FOR K:=0 TO 5 DO
008A 3-                  BEGIN
008A --                      FOR L:=0 TO 5 DO
00B2 4-                      BEGIN
00B2 --                        Z(.I*216+J*36+K*6+L.):=SQR(((2*I-5)*X(.1.)+(2*J-5)*X(.2.)
0150 --                        +(2*K-5)*X(.3.)+(2*L-5)*X(.4.)));
01E8 -4                      END;
021C -3                  END;
0250 -2             END;
0284 -1          END;
02B8 --          V(.1.):=0;
02D2 --          FOR I:=0 TO 1295 DO V(.1.):=V(.1.)+Z(.I.);
0370 --          FOR J:=2 TO FIN DO
039A 1-          BEGIN
039A --                  V(.J.):=0;
03D0 --                  FOR I:=0 TO 1295 DO
03F8 2-                  BEGIN
03F8 --                  IF (LN(Z(.I.))/LN10)*J > -70 THEN
0460 --                      V(.J.):=V(.J.)+Z(.I.)**J;
```

```
04EE -2               END;
0522 -1           END;
0556 --           FOR I:=1 TO FIN DO V(.I.):=V(.I.)/1296;
061C -0 A     END;
0658 --
0658 --       (*-----------------------------------------------------------------------*)
0658 --
0658 -- A     PROCEDURE MOM64(VAR V:VECTEUR;X1,X2:REAL);
0058 --       (*POUR CALCULER LES MOMENTS NORMALISES A PARTIR
0058 --         DE DEUX ECHANTILLONS, *)
0058 --       VAR
0058 --           I,J:INTEGER;
0060 --           Z:ARRAY(.0..63.) OF REAL;
0000 0- A     BEGIN
0012 --           FOR I:=0 TO 7 DO
002E 1-           BEGIN
002E --               FOR J:=0 TO 7 DO
004A 2-               BEGIN
004A --                   Z(.8*I+J.):=SQR(((2*I-7)*X1+(2*J-7)*X2));
00D0 -2               END;
00F2 -1           END;
0114 --           FOR J:=1 TO FIN DO
0132 1-           BEGIN
0132 --               V(.J.):=0;
0162 --               FOR I:=0 TO 63 DO
017E 2-               BEGIN
017E --                   IF (LN(Z(.I.))/LN10)*J > -70 THEN
01DA --                       V(.J.):=V(.J.)+Z(.I.)**J;
0250 -2               END;
0272 --               V(.J.):=V(.J.)/64;
02CE -1           END;
02F0 -0 A     END;
0318 --
0318 --       (*-----------------------------------------------------------------------*)
0318 --
0318 -- A     PROCEDURE MOM4(VAR V:VECTEUR;X:VECT);
0088 --       (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --       VAR
0088 --           Z:ARRAY(.0..255.) OF REAL;
0888 --           I,J,K,L,M,N,P,Q:INTEGER;
0000 0- A     BEGIN
0012 --        FOR I:=0 TO 1 DO
002E 1-        BEGIN
002E --         FOR J:=0 TO 1 DO
004A 2-         BEGIN
004A --          FOR K:=0 TO 1 DO
0066 3-          BEGIN
0066 --           FOR L:=0 TO 1 DO
0082 4-           BEGIN
0082 --           FOR M:=0 TO 1 DO
009E 5-           BEGIN
009E --            FOR N:=0 TO 1 DO
00BA 6-            BEGIN
00BA --             FOR P:=0 TO 1 DO
00D6 7-             BEGIN
00D6 --              FOR Q:=0 TO 1 DO
00F2 8-              BEGIN
00F2 --                  Z(.I*128+J*64+K*32+L*16+M*8+N*4+P*2+Q.):=
```

```
0164 --                SQR(((2*I-1)*X(.1.)+(2*J-1)*X(.2.)
01A4 --                +(2*K-1)*X(.3.)+(2*L-1)*X(.4.)+(2*M-1)*X(.5.)
023A --                +(2*N-1)*X(.6.)+(2*P-1)*X(.7.)+(2*Q-1)*X(.8.)));
02F8 -8            END;
031A -7           END;
033C -6          END;
035E -5         END;
0380 -4        END;
03A2 -3       END;
03C4 -2      END;
03E6 -1     END;
0408 --       V(.1.):=0;
0422 --       FOR I:=0 TO 255 DO V(.1.):=V(.1.)+Z(.I.);
049C --       FOR J:=2 TO FIN DO
04BA 1-       BEGIN
04BA --                V(.J.):=0;
04EA --                FOR I:=0 TO 255 DO
0506 2-                BEGIN
0506 --                IF (LN(Z(.I.))/LN10)*J > -70 THEN
0562 --                    V(.J.):=V(.J.)+Z(.I.)**J;
05D8 -2                END;
05FA -1       END;
061C --       FOR I:=1 TO FIN DO V(.I.):=V(.I.)/256;
06B8 -0  A  END;
06F8 --
06F8 --     (*-----------------------------------------------------------------*)
06F8 --
06F8 --  A  FUNCTION COMB(X:REAL;K,N:INTEGER):REAL;
0058 --     (*POUR CALCULER LES COMBINAISONS "K PARMI N".*)
0058 --     VAR
0058 --        I:INTEGER;
0000 0-  A  BEGIN
0018 --        FOR I:=N DOWNTO K+1 DO
004C 1-            BEGIN
004C --                X:=X*I/(I-K);
0096 -1            END;
00B8 --        COMB:=X;
00CA -0  A  END;
00E8 --
00E8 --     (*-----------------------------------------------------------------*)
00E8 --
00E8 --  A  PROCEDURE UNION(VAR X,Y,Z:VECTEUR);
004C --     (*POUR COMBINER DEUX MOMENTS.*)
004C --     VAR
004C --        I,J:INTEGER;
0054 --        B,C:REAL;
0000 0-  A  BEGIN
0012 --        FOR I:=1 TO FIN DO Z(.I.):=X(.I.)+Y(.I.);
00C2 --        FOR I:=2 TO FIN DO
00E0 1-            BEGIN
00E0 --                J:=1;
00E8 2-                REPEAT
00E8 --                IF (X(.J.)<>0) AND (Y(.I-J.)<>0) THEN
0188 3-                BEGIN
0188 --                   B:=LN(Y(.I-J.));
01C8 --                   C:=LN(COMB(X(.J.),2*J,2*I));
0230 --                   IF (B+C)/LN10>-70 THEN
026C --                   Z(.I.):=Z(.I.)+EXP(B+C);
```

```
02DC -3                  END;
02DC --                  J:=J+1
02E6 -2                  UNTIL J>=I;
0300 -1              END;
0322 -0 A   END;
034C --
034C --     (*----------------------------------------------------------------------*)
034C --
034C -- A   FUNCTION ERFC(VAR X:REAL):REAL;
004C --     (*CALCUL DE LA FONCTION D'ERREUR COMPLEMENTAIRE PAR LA METHODE
004C --     DE GAUSS D'ORDRE 20*)
004C --
004C --     TYPE
004C --         GAUSS=ARRAY(.1..10.) OF REAL;
004C --     VAR
004C --         T,W:GAUSS;
00F0 --         A,B,R,C:REAL;
0110 --         I:INTEGER;
0114 --
0114 -- B   FUNCTION F(VAR X:REAL):REAL;
004C --     CONST PI=3.14159265358979;
0000 0- B   BEGIN
0018 --         F:=(2/SQRT(PI))*EXP(-X*X);
0072 -0 B   END;
0094 --
0000 0- A   BEGIN
0018 --         T(.1.):=0.076526521133497;W(.1.):=0.152753387130725;
0028 --         T(.2.):=0.227785851141645;W(.2.):=0.149172986472603;
0038 --         T(.3.):=0.373706088715419;W(.3.):=0.142096109318382;
0048 --         T(.4.):=0.510867001950827;W(.4.):=0.131688638449176;
0058 --         T(.5.):=0.636053680726515;W(.5.):=0.118194531961518;
0068 --         T(.6.):=0.746331906460150;W(.6.):=0.101930119817240;
0078 --         T(.7.):=0.839116971822218;W(.7.):=0.083276741576704;
0088 --         T(.8.):=0.912234428251325;W(.8.):=0.062672048334109;
0098 --         T(.9.):=0.963971927277913;W(.9.):=0.040601429800386;
00A8 --         T(.10.):=0.993128599185094;W(.10.):=0.017614007139152;
00B8 --         IF ABS(X)>12 THEN R:=0 ELSE
0100 1-             BEGIN
0100 --             A:=(13-ABS(X))/2;
0144 --             B:=(13+ABS(X))/2;
0188 --             R:=0;
019E --             FOR I:=1 TO 10 DO
01BC 2-                 BEGIN
01BC --                 C:=B+A*T(.I.);
0200 --                 R:=R+W(.I.)*F(C);
0252 -2                 END;
0274 --             FOR I:=1 TO 10 DO
0292 2-                 BEGIN
0292 --                 C:=B-A*T(.I.);
02D6 --                 R:=R+W(.I.)*F(C);
0328 -2                 END;
034A -1             END;
034A --         IF X<0 THEN ERFC:=2-A*R ELSE ERFC:=A*R;
03D2 -0 A   END;
0494 --
0494 --     (*----------------------------------------------------------------------*)
0494 --
0494 -- A   PROCEDURE MOMBS(VAR V:VECTEUR;S:R:REAL);
```

```
0050 --      (*POUR CALCULER LES MOMENTS NORMALISES D'UN
0050 --        BROUILLEUR SINUSOIDAL. *)
0050 --      VAR
0050 --         I:INTEGER;
0054 --         ARG,S:REAL;
0000 0-  A   BEGIN
0012 --         ARG:=10**(-SIR/10);
0064 --           ARG:=ARG*2*(SQR(2*N)-1)/3;
00D6 --         V(.1.):=0,5*ARG;
00F0 --          FOR I:=2 TO FIN DO
010E 1-               BEGIN
010E --               IF V(.I-1.)<>0 THEN
0152 2-                    BEGIN
0152 --                    S:=LN(V(.I-1.))+LN(ARG*(2*I-1)/(2*I));
01E4 --                    IF S/LN10>-70 THEN V(.I.):=EXP(S);
0244 -2                    END;
0244 -1               END;
0266 -0  A   END;
0298 --
0298 --      (*-----------------------------------------------------------------------*)
0298 --
0000 0-      BEGIN
003A --         CNOB:=18;
0058 --           WRITELN('                 FILTRE BUTTERWORTH ORDRE 5');
0076 --          WRITELN;
0082 --          WRITELN('                 BROUILLEUR SINUSOIDAL');
00A0 --         WRITELN;WRITELN;
00B8 --         FOR Q:=0 TO 0 DO
00E0 1-         BEGIN
00E0 --         BT:=1,05+0,03*Q;
010C --         WRITELN('        BT=',BT:5:2);
0150 --      WRITELN;WRITELN;
0168 --      X1:=ABS(REP(BT,0));
019A --          FOR N:=2 TO 2 DO
01C4 2-          BEGIN
01C4 --          WRITELN('        MODULATION QAM=',SQR(2*N):3);
020C --      WRITELN;WRITELN;
0224 3-         CASE N OF
0232 4-             1: BEGIN
0232 --                FOR I:=1 TO 8 DO VARIAB(.I.):=REP(BT,I)/X1;
02F0 --                 MOM4(VECT1,VARIAB);
0304 --                UNION(VECT1,VECT1,RES1);
0326 -4                END;
032A 4-             2: BEGIN
032A --                FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I)/X1;
03F0 --                MOM16(RES1,VARIAB);
0404 --                FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I-1)/X1;
04CE --                MOM16(RES2,VARIAB);
04E2 --                UNION(RES1,RES2,VECT1);
0504 --                UNION(VECT1,VECT1,RES1);
0526 -4                END;
052A 4-             3: BEGIN
052A --               FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I)/X1;
05F0 --                MOM36(RES1,VARIAB);
0604 --                FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I-1)/X1;
06CE --                MOM36(RES2,VARIAB);
06E2 --                UNION(RES1,RES2,VECT1);
0704 --                UNION(VECT1,VECT1,RES1);
```

```
0726 -4              END;
072A 4-          4: BEGIN
072A --              FOR I:=1 TO 8 DO VARIAB(,I,):=REP(BT,I)/X1;
07E8 --              MOM64(RES1,VARIAB(,1,),VARIAB(,8,));
081A --              MOM64(RES2,VARIAB(,2,),VARIAB(,7,));
084C --              UNION(RES1,RES2,VECT1);
086E --              MOM64(RES1,VARIAB(,3,),VARIAB(,6,));
08A0 --              MOM64(RES2,VARIAB(,4,),VARIAB(,5,));
08D2 --              UNION(RES1,RES2,VECT2);
08F4 --              UNION(VECT1,VECT2,RES3);
0914 --              UNION(RES3,RES3,RES1);
0932 -4              END
0932 -3          END;
095E --      FOR L:=0 TO 0 DO
0986 3-      BEGIN
0986 --      SIR:=20-2*L;
09B8 --      WRITELN('        SIR=',SIR:5:1,' DB');
0A0E --  WRITELN;
0A1A --      MOMBS(RES2,SIR);
0A40 --      UNION(RES1,RES2,VECT1);
0A62 --     FOR J:=0 TO 12 DO
0A8A 4-     BEGIN
0A8A --          CNO:=J+CNOB;
0AC4 --          WRITE('     SNR=',CNO:5:1,' DB');
0B14 --          SIGMA:=X1*SQRT(3*(10**(CNO/10))/(2*(SQR(2*N)-1)*BT*KT));
0BF6 --          SIGMA2:=(SIGMA/X1)*SQRT(BT*KT);
0C4E --          PE1:=(2-1/N)*0,5*ERFC(SIGMA2);
0CC8 --          WRITE('      PE1=',PE1:9);
0D04 --          PE2:=(2-1/N)*(0,5*ERFC(SIGMA)+HERMITE(SIGMA,RES1));
0DC6 --          WRITE('   TERMES          PE2=',PE2:9);
0E02 --          PE3:=(2-1/N)*(0,5*ERFC(SIGMA)+HERMITE(SIGMA,VECT1));
0EC4 --          WRITELN('   TERMES          PE3=',PE3:9);
0F06 -4      END;
0F3A --     WRITELN;
0F46 -3    END;
0F7A --      WRITELN;
0F86 -2  END;
0FBA --      WRITELN;
0FC6 -1  END;
0FFA -0  5: END.
```

PROGRAMME QAMFM

Calcule la probabilité d'erreur moyenne à la sortie d'un récepteur QAM avec ou sans IIS en présence d'une interférence FM. L'interférence FM considérée ici s'appuie sur l'hypothèse de quasi-stationnarité qui implique que le signal de modulation FM et à très basse fréquence (modulation à largeur de bande étroite). Le canal est modélisé pour les besoins du calcul de l'IIS par un filtre Butterworth d'ordre 5 et d'une "pulse" de transmission rectangulaire. Pour le calcul de l'interférence FM, on considère toutefois que le filtre de réception est idéal pour les raisons mentionnées dans le rapport. Le calcul des moments IIS est réalisé comme précédemment par une méthode directe modifiée.

Lignes

1-367	Voir QAMBS	
371-418	comme précédemment, on calcule les échantillons du canal et les moments IIS d'ordre 2 à 2K (K = FIN). Les moments IIS sont dans RES1.	
423	on calcule les moments de l'interférence FM considérée constante par l'hypothèse de quasi-stationnarité. Le résultat est dans RES2.	
424	on combine les moments IIS avec les moments FM. Le résultat est dans VECT1.	
431	on calcule P_e sans IIS et sans interférence	PE1
433	on calcule P_e sans IIS avec interférence sinusoïdale constante	PE2
435	on calcule P_e avec IIS et interférence	PE3
437	on calcule P_e avec IIS mais sans interférence	PE4

449 on calcule P_k, la probabilité que la fréquence instantanée de la porteuse FM soit à l'intérieur de la bande du récepteur.

X : écart de fréquence entre les porteuses QAM et FM normalisé à B/2 la fréquence de coupure du filtre de réception.

Y : déviation de fréquence RMS normalisée à B/2 (écart type du signal de modulation FM de type gaussien)

451 on calcule P_e avec IIS pondéré avec P_k PE5

452 on calcule P_e sans IIS pondéré par P_k PE6

```
0680 --      (*+L+*)
0680 --      PROGRAM QAMFM(OUTPUT);
0694 --      LABEL 5;
0694 --      CONST
0694 --          Z=5;
0694 --         FIN=200;
0694 --         PI2=6.2831853071795862;
0694 --          KT=1.016640738;
0694 --         LN10=2.302585092994045б;
0694 --      TYPE
0694 --         VECT=ARRAY(.1..8.) OF REAL;
0694 --         VECTEUR=ARRAY(.1..FIN.) OF REAL;
0694 --          SPEC=ARRAY(.1..2,1..15.) OF REAL;
0694 --      VAR
0694 --         VARIAB:VECT;
06D8 --         VECT1,VECT2,RES1,RES2,RES3,RES4:VECTEUR;
2C58 --         I,J,L,N,K,M,POSI,PAS,P,Q,R,V,S:INTEGER;
2C8C --         CNOB,CNO,SIGMA,SIGMA2,BT,X1,SIR:REAL;
2CC8 --          A,B,X,PE1,Y,PK,PE2,PE3,PE4,PE5,PE6:REAL;
2D20 --
2D20 --      (*----------------------------------------------------------------------*)
2D20 --
2D20 --  A   FUNCTION FREQ(X,BT:REAL;K:INTEGER):REAL;
005C --      (*CALCUL DE LA FONCTION A INTEGRER*)
005C --      VAR Y,S:REAL;
0000 0-  A   BEGIN
0018 --          IF X=0 THEN FREQ:=2
003E 1-                 ELSE BEGIN Y:=2*(COS(PI2*X*K))*SIN(PI2*X/2)/(PI2*X/2);
0108 --                             S:=2*Z*LN(2*X/BT);
0166 --                             IF ABS(S)>70 THEN
0190 --                                      IF X<(BT/2) THEN FREQ:=Y
01C8 --                                                   ELSE FREQ:=0
01DE --                                      ELSE FREQ:=Y/SQRT(1+(2*X/BT)**(2*Z));
0276 -1                       END;
0276 -0  A   END;
029C --
029C --      (*----------------------------------------------------------------------*)
029C --
029C --  A   FUNCTION ROM(BT,BS,BI:REAL;K:INTEGER):REAL;
0064 --      (*CALCUL DE LA REPONSE DU CANAL*)
0064 --      LABEL 5;
0064 --      VAR
0064 --          H,L:REAL;
0078 --          I,J,P:INTEGER;
0084 --          R:SPEC;
0000 0-  A   BEGIN
0018 --          H:=BS-BI;
003A --          R(.1,1.):=H*(FREQ(BI,BT,K)+FREQ(BS,BT,K))/2;
00E6 --          I:=1;
00EE 1-          REPEAT
00EE --          I:=I+1;
0100 --                      L:=0;
0116 --                      FOR P:=1 TO 2**(I-2) DO
0154 2-                           BEGIN
0154 --                           L:=L+FREQ(BI+((2*P-1)*H/2),BT,K);
01F0 -2                           END;
0212 --                      R(.2,1.):=(R(.1,1.)+H*L)/2;
025A --                      FOR J:=2 TO I DO
```

```
0286 2-                         BEGIN
0286 --                         R(.2,J.):=R(.2,J-1.)*(4**(J-1))-R(.1,J-1.);
030E --                         R(.2,J.):=R(.2,J.)/(4**(J-1)-1);
0370 -2                         END;
0392 --                   H:=H/2;
03BA --                   FOR J:=1 TO I DO R(.1,J.):=R(.2,J.);
043E --          IF R(.2,I.)=0 THEN GOTO 5;
047A -1          UNTIL (ABS((R(.2,I.)-R(.2,I-1.))/R(.2,I.))<1E-12) OR (I=15);
0518 --          5:ROM:=R(.2,I.);
053C -0 A    END;
056C --
056C --      (*------------------------------------------------------------------*)
056C --
056C -- A      FUNCTION REP(BT:REAL;K:INTEGER):REAL;
0054 --      VAR
0054 --           I:INTEGER;
0058 --           S:REAL;
0000 0- A    BEGIN
0018 --           S:=0;
002E --         FOR I:=0 TO 12 DO S:=S+ROM(BT,I+1,I,K);
00E2 --         REP:=S;
00F4 -0 A    END;
0114 --
0114 --
0114 --      (*------------------------------------------------------------------*)
0114 --
0114 --
0114 -- A    FUNCTION HERMITE(X:REAL;Z:VECTEUR):REAL;
0690 --      VAR
0690 --          J:INTEGER;
0694 --          ARG,TRANS,A,B:REAL;
06B8 --          V:VECTEUR;
0000 0- A    BEGIN
0018 --          ARG:=(1/SQRT(PI2/2))*EXP(-X*X);
0082 --          V(.1.):=X**3;
00A2 --          V(.2.):=(X**4)*(8*(X**3)-12*X)/24;
0132 --          J:=2;
013A --          TRANS:=V(.1.)*Z(.1.)+V(.2.)*Z(.2.);
017C 1-          REPEAT
017C --                J:=J+1;
018E --                A:=(1-((4*J-5)/(2*X*X)))*V(.J-1.);
0232 --                B:=V(.J-2.)*(J-2)/(J-1);
02A0 --                V(.J.):=(2*(X**4)/(J*(2*J-1)))*(A-B);
0344 --                TRANS:=TRANS+V(.J.)*Z(.J.);
039A -1          UNTIL (ABS(V(.J.)*Z(.J.)/TRANS)<1E-5) OR (J=FIN);
041C --          HERMITE:=TRANS*ARG;
043E --          WRITE('      ',J:3);
046E -0 A    END;
04B4 --
04B4 --
04B4 --      (*------------------------------------------------------------------*)
04B4 --
04B4 -- A    PROCEDURE MOM16(VAR V:VECTEUR;X:VECT);
0088 --      (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --      VAR
0088 --         Z:ARRAY(.0..255.) OF REAL;
0888 --         I,J,K,L:INTEGER;
0000 0- A    BEGIN
```

```
0012 --        FOR I:=0 TO 3 DO
002E 1-        BEGIN
002E --             FOR J:=0 TO 3 DO
004A 2-             BEGIN
004A --                  FOR K:=0 TO 3 DO
0066 3-                  BEGIN
0066 --                       FOR L:=0 TO 3 DO
0082 4-                       BEGIN
0082 --                         Z(.I*64+J*16+K*4+L.):=SQR(((2*I-3)*X(.1.)+(2*J-3)*X(.2.)
00FC --                         +(2*K-3)*X(.3.)+(2*L-3)*X(.4.)));
0188 -4                       END;
01AA -3                  END;
01CC -2              END;
01EE -1        END;
0210 --        V(.1.):=0;
022A --        FOR I:=0 TO 255 DO V(.1.):=V(.1.)+Z(.I.);
02A4 --        FOR J:=2 TO FIN DO
02C2 1-        BEGIN
02C2 --                  V(.J.):=0;
02F2 --                  FOR I:=0 TO 255 DO
030E 2-                  BEGIN
030E --                  IF (LN(Z(.I.))/LN10)*J > -70 THEN
036A --                       V(.J.):=V(.J.)+Z(.I.)**J;
03E0 -2                  END;
0402 -1        END;
0424 --        FOR I:=1 TO FIN DO V(.I.):=V(.I.)/256;
04C0 -0 A  END;
04F4 --
04F4 --    (*-----------------------------------------------------------------------*)
04F4 --
04F4 -- A  PROCEDURE MOM36(VAR V:VECTEUR;X:VECT);
0088 --    (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --    VAR
0088 --        Z:ARRAY(.0..1295.) OF REAL;
2908 --        I,J,K,L:INTEGER;
0000 0- A  BEGIN
0012 --        FOR I:=0 TO 5 DO
003A 1-        BEGIN
003A --             FOR J:=0 TO 5 DO
0062 2-             BEGIN
0062 --                  FOR K:=0 TO 5 DO
008A 3-                  BEGIN
008A --                       FOR L:=0 TO 5 DO
0082 4-                       BEGIN
0082 --                         Z(.I*216+J*36+K*6+L.):=SQR(((2*I-5)*X(.1.)+(2*J-5)*X(.2.)
0150 --                         +(2*K-5)*X(.3.)+(2*L-5)*X(.4.)));
01E8 -4                       END;
021C -3                  END;
0250 -2              END;
0284 -1        END;
0288 --        V(.1.):=0;
02D2 --        FOR I:=0 TO 1295 DO V(.1.):=V(.1.)+Z(.I.);
0370 --        FOR J:=2 TO FIN DO
039A 1-        BEGIN
039A --                  V(.J.):=0;
03D0 --                  FOR I:=0 TO 1295 DO
03F8 2-                  BEGIN
03F8 --                  IF (LN(Z(.I.))/LN10)*J > -70 THEN
```

```
0460 --                      V(.J.):=V(.J.)+Z(.I.)**J;
04EE -2                  END;
0522 -1          END;
0556 --          FOR I:=1 TO FIN DO V(.I.):=V(.I.)/1296;
061C -0 A    END;
0658 --
0658 --      (*--------------------------------------------------------------*)
0658 --
0658 -- A    PROCEDURE MOM64(VAR V:VECTEUR;X1,X2:REAL);
0058 --      (*POUR CALCULER LES MOMENTS NORMALISES A PARTIR
0058 --        DE DEUX ECHANTILLONS. *)
0058 --      VAR
0058 --          I,J:INTEGER;
0060 --          Z:ARRAY(.0..63.) OF REAL;
0000 0- A    BEGIN
0012 --          FOR I:=0 TO 7 DO
002E 1-          BEGIN
002E --              FOR J:=0 TO 7 DO
004A 2-              BEGIN
004A --                  Z(.8*I+J.):=SQR(((2*I-7)*X1+(2*J-7)*X2));
00D0 -2              END;
00F2 -1          END;
0114 --          FOR J:=1 TO FIN DO
0132 1-          BEGIN
0132 --              V(.J.):=0;
0162 --              FOR I:=0 TO 63 DO
017E 2-              BEGIN
017E --                  IF (LN(Z(.I.))/LN10)*J > -70 THEN
01DA --                      V(.J.):=V(.J.)+Z(.I.)**J;
0250 -2              END;
0272 --              V(.J.):=V(.J.)/64;
02CE -1          END;
02F0 -0 A    END;
0318 --
0318 --      (*--------------------------------------------------------------*)
0318 --
0318 -- A    PROCEDURE MOM4(VAR V:VECTEUR;X:VECT);
0088 --      (*POUR CALCULER LES MOMENTS A PARTIR DE 4 ECHANTILLONS.*)
0088 --      VAR
0088 --          Z:ARRAY(.0..255.) OF REAL;
0888 --          I,J,K,L,M,N,P,Q:INTEGER;
0000 0- A    BEGIN
0012 --       FOR I:=0 TO 1 DO
002E 1-       BEGIN
002E --        FOR J:=0 TO 1 DO
004A 2-        BEGIN
004A --         FOR K:=0 TO 1 DO
0066 3-         BEGIN
0066 --          FOR L:=0 TO 1 DO
0082 4-          BEGIN
0082 --          FOR M:=0 TO 1 DO
009E 5-          BEGIN
009E --           FOR N:=0 TO 1 DO
00BA 6-           BEGIN
00BA --            FOR P:=0 TO 1 DO
00D6 7-            BEGIN
00D6 --             FOR Q:=0 TO 1 DO
00F2 8-             BEGIN
```

```
00F2 --                    Z(.I*128+J*64+K*32+L*16+M*8+N*4+P*2+Q,):=
0164 --                    SQR(((2*I-1)*X(.1.)+(2*J-1)*X(.2.)
01A4 --                    +(2*K-1)*X(.3.)+(2*L-1)*X(.4.)+(2*M-1)*X(.5.)
023A --                    +(2*N-1)*X(.6.)+(2*P-1)*X(.7.)+(2*Q-1)*X(.8.)));
02F8 -8                 END;
031A -7                END;
033C -6               END;
035E -5              END;
0380 -4             END;
03A2 -3            END;
03C4 -2           END;
03E6 -1          END;
0408 --            V(.1.):=0;
0422 --            FOR I:=0 TO 255 DO V(.1.):=V(.1.)+Z(.I.);
049C --            FOR J:=2 TO FIN DO
04BA 1-            BEGIN
04BA --                    V(.J.):=0;
04EA --                    FOR I:=0 TO 255 DO
0506 2-                    BEGIN
0506 --                    IF (LN(Z(.I.))/LN10)*J > -70 THEN
0562 --                        V(.J.):=V(.J.)+Z(.I.)**J;
05D8 -2                    END;
05FA -1            END;
061C --            FOR I:=1 TO FIN DO V(.I.):=V(.I.)/256;
06B8 -0 A     END;
06F8 --
06F8 --       (*-----------------------------------------------------------------*)
06F8 --
06F8 -- A     FUNCTION COMB(X:REAL;K,N:INTEGER):REAL;
0058 --       (*POUR CALCULER LES COMBINAISONS "K PARMI N",*)
0058 --       VAR
0058 --          I:INTEGER;
0000 0- A     BEGIN
0018 --          FOR I:=N DOWNTO K+1 DO
004C 1-              BEGIN
004C --                  X:=X*I/(I-K);
0096 -1              END;
00B8 --          COMB:=X;
00CA -0 A     END;
00E8 --
00E8 --       (*-----------------------------------------------------------------*)
00E8 --
00E8 -- A     PROCEDURE UNION(VAR X,Y,Z:VECTEUR);
004C --       (*POUR COMBINER DEUX MOMENTS,*)
004C --       VAR
004C --          I,J:INTEGER;
0054 --          B,C:REAL;
0000 0- A     BEGIN
0012 --          FOR I:=1 TO FIN DO Z(.I.):=X(.I.)+Y(.I.);
00C2 --          FOR I:=2 TO FIN DO
00E0 1-              BEGIN
00E0 --                  J:=1;
00E8 2-                  REPEAT
00E8 --                  IF (X(.J.)<>0) AND (Y(.I-J.)<>0) THEN
0188 3-                  BEGIN
0188 --                     B:=LN(Y(.I-J.));
01C8 --                     C:=LN(COMB(X(.J.),2*J,2*I));
0230 --                     IF (B+C)/LN10>=70 THEN
```

```
026C --                Z(.I.):=Z(.J.)+EXP(B+C);
02DC -3             END;
02DC --             J:=J+1
02E6 -2             UNTIL J>=I;
0300 -1         END;
0322 -0 A   END;
034C --
034C --     (*-----------------------------------------------------------------------*)
034C --
034C -- A   FUNCTION ERFC(VAR X:REAL):REAL;
004C --     (*CALCUL DE LA FONCTION D'ERREUR COMPLEMENTAIRE PAR LA METHODE
004C --     DE GAUSS D'ORDRE 20*)
004C --
004C --     TYPE
004C --         GAUSS=ARRAY(.1..10.) OF REAL;
004C --     VAR
004C --         T,W:GAUSS;
00F0 --         A,B,R,C:REAL;
0110 --         I:INTEGER;
0114 --
0114 -- B   FUNCTION F(VAR X:REAL):REAL;
004C --     CONST PI=3.14159265358979;
0000 0- B   BEGIN
0018 --         F:=(2/SQRT(PI))*EXP(-X*X);
0072 -0 B   END;
0094 --
0000 0- A   BEGIN
0018 --         T(.1.):=0.076526521133497;W(.1.):=0.152753387130725;
0028 --         T(.2.):=0.227785851141645;W(.2.):=0.149172986472603;
0038 --         T(.3.):=0.373706088715419;W(.3.):=0.142096109318382;
0048 --         T(.4.):=0.510867001950827;W(.4.):=0.131688638449176;
0058 --         T(.5.):=0.636053680726515;W(.5.):=0.118194531961518;
0068 --         T(.6.):=0.746331906460150;W(.6.):=0.101930119817240;
0078 --         T(.7.):=0.839116971822218;W(.7.):=0.083276741576704;
0088 --         T(.8.):=0.912234428251325;W(.8.):=0.062672048334109;
0098 --         T(.9.):=0.963971927277913;W(.9.):=0.040601429800386;
00A8 --         T(.10.):=0.993128599185094;W(.10.):=0.017614007139152;
00B8 --         IF ABS(X)>12 THEN R:=0 ELSE
0100 1-             BEGIN
0100 --             A:=(13-ABS(X))/2;
0144 --             B:=(13+ABS(X))/2;
0188 --             R:=0;
019E --             FOR I:=1 TO 10 DO
01BC 2-                 BEGIN
01BC --                 C:=B+A*T(.I.);
0200 --                 R:=R+W(.I.)*F(C);
0252 -2                 END;
0274 --             FOR I:=1 TO 10 DO
0292 2-                 BEGIN
0292 --                 C:=B-A*T(.I.);
02D6 --                 R:=R+W(.I.)*F(C);
0328 -2                 END;
034A -1             END;
034A --         IF X<0 THEN ERFC:=2-A*R ELSE ERFC:=A*R;
03D2 -0 A   END;
0494 --
0494 --     (*-----------------------------------------------------------------------*)
0494 --
```

```
0494 -- A   PROCEDURE MOMBS(VAR V:VECTEUR;SIR:REAL);
0050 --     (*POUR CALCULER LES MOMENTS NORMALISES D'UN
0050 --       BROUILLEUR SINUSOIDAL. *)
0050 --     VAR
0050 --        I:INTEGER;
0054 --        ARG,S:REAL;
0000 0- A   BEGIN
0012 --        ARG:=10**(-SIR/10);
0064 --          ARG:=ARG*2*(SQR(2*N)-1)/3;
0006 --        V(.1.):=0.5*ARG;
00F0 --         FOR I:=2 TO FIN DO
010E 1-              BEGIN
010E --              IF V(.I-1.)<>0 THEN
0152 2-                   BEGIN
0152 --                   S:=LN(V(.I-1.))+LN(ARG*(2*I-1)/(2*I));
01E4 --                   IF S/LN10>=70 THEN V(.I.):=EXP(S);
0244 -2                   END;
0244 -1              END;
0266 -0 A   END;
0298 --
0298 --     (*----------------------------------------------------------------------*)
0298 --
0000 0-     BEGIN
003A --        CNDB:=18;
0058 --           WRITELN('                       FILTRE BUTTERWORTH ORDRE 5');
0076 --          WRITELN;
0082 --          WRITELN('                        BROUILLEUR FM');
0040 --        WRITELN;WRITELN;
0088 --        FOR Q:=0 TO 0 DO
00E0 1-        BEGIN
00E0 --        BT:=1.05+0.03*Q;
010C --        WRITE('       BT=',BT:5:2);
014A --     X1:=ABS(REP(BT,0));
017C --         FOR N:=2 TO 2 DO
01A6 2-         BEGIN
01A6 --         WRITELN('           MODULATION QAM=',SQR(2*N):3);
01EE 3-        CASE N OF
01FC 4-             1: BEGIN
01FC --                FOR I:=1 TO 8 DO VARIAB(.I.):=REP(BT,I)/X1;
02BA --                 MOM4(VECT1,VARIAB);
02CE --                UNION(VECT1,VECT1,RES1);
02F0 -4                END;
02F4 4-             2: BEGIN
02F4 --                FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I)/X1;
03BA --                MOM16(RES1,VARIAB);
03CE --                FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I-1)/X1;
0498 --                MOM16(RES2,VARIAB);
04AC --                UNION(RES1,RES2,VECT1);
04CE --                UNION(VECT1,VECT1,RES1);
04F0 -4                END;
04F4 4-             3: BEGIN
04F4 --               FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I)/X1;
05BA --                MOM36(RES1,VARIAB);
05CE --                FOR I:=1 TO 4 DO VARIAB(.I.):=REP(BT,2*I-1)/X1;
0698 --                MOM36(RES2,VARIAB);
06AC --                UNION(RES1,RES2,VECT1);
06CE --                UNION(VECT1,VECT1,RES1);
06F0 -4                END;
```

```
06F4 4-            4: BEGIN
06F4 --               FOR I:=1 TO 8 DO VARIAB(.I.):=REP(BT,I)/X1;
07B2 --               MOM64(RES1,VARIAB(.1.),VARIAB(.8.));
07E4 --               MOM64(RES2,VARIAB(.2.),VARIAB(.7.));
0816 --               UNION(RES1,RES2,VECT1);
0838 --               MOM64(RES1,VARIAB(.3.),VARIAB(.6.));
086A --               MOM64(RES2,VARIAB(.4.),VARIAB(.5.));
089C --               UNION(RES1,RES2,VECT2);
08BE --               UNION(VECT1,VECT2,RES3);
08DE --               UNION(RES3,RES3,RES1);
08FC -4               END
08FC -3            END;
0928 --         FOR L:=0 TO 0 DO
0950 3-         BEGIN
0950 --         SIR:=20-2*L;
0982 --         WRITE('       SIR=',SIR:5:1,' DB');
09D2 --         MOMBS(RES2,SIR);
09F8 --         UNION(RES1,RES2,VECT1);
0A1A --        FOR J:=0TO 12 DO
0A42 4-        BEGIN
0A42 --            CNO:=J+CNOB;
0A7C --            WRITELN('      SNR=',CNO:5:1,' DB');
0AD2 --            SIGMA:=X1*SQRT(3*(10**(CNO/10))/(2*(SQR(2*N)-1)*BT*KT));
0BB4 --            SIGMA2:=(SIGMA/X1)*SQRT(BT*KT);
0C0C --            PE1:=(2-1/N)*0,5*ERFC(SIGMA2);
0C86 --            WRITELN('         PE1=',PE1:9);
0CC8 --            PE2:=(2-1/N)*(0.5*ERFC(SIGMA2)+HERMITE(SIGMA2,RES2));
0D8A --            WRITELN('    TERMES          PE2=',PE2:9);
0DCC --            PE3:=(2-1/N)*(0.5*ERFC(SIGMA)+HERMITE(SIGMA,VECT1));
0E8E --            WRITELN('    TERMES          PE3=',PE3:9);
0ED0 --            PE4:=(2-1/N)*(0.5*ERFC(SIGMA)+HERMITE(SIGMA,RES1));
0F92 --            WRITELN('    TERMES          PE4=',PE4:9);
0FD4 --              FOR K:=1 TO 5 DO
0FFE 5-              BEGIN
0FFE --                   Y:=K;
1022 --                   WRITELN('        DISPERSION FM',Y:9:2);
1066 --                   FOR I:=0 TO 10 DO
108E 6-                   BEGIN
108E --                        X:=I*0,5;
10B6 --                        WRITE('        POSITION FM',X:7:2);
10F4 --                        A:=(X-1)/(Y*SQRT(2));
115E --                        B:=(X+1)/(Y*SQRT(2));
11C8 --                        PK:=0,5*(ERFC(A)-ERFC(B));
1210 --                        WRITE('      PK=',PK:9);
124C --                        PE5:=PE4+PK*(PE3-PE4);
12AC --                        PE6:=PE1+PK*(PE2-PE1);
130C --                        WRITE('          PE5=',PE5:9);
1348 --                        WRITELN('            PE6=',PE6:9);
138A -6                   END;
13BE --              WRITELN;
13CA -5              END;
13FE --              WRITELN('1');
143C -4         END;
1470 --        WRITELN;
147C -3       END;
14B0 --         WRITELN;
14BC -2     END;
14F0 --         WRITELN;
```

```
14FC -1      END;
1530 -0      5; END.
```

PROGRAMME QAMBSN

Permet le calcul de P_e dans un récepteur QAM en absence d'IIS mais en présence de plusieurs interférences sinusoïdales constantes, de même amplitude dont les phases sont également réparties et indépendantes. La puissance totale des interférences est constante quand on varie leur nombre par le paramètre I.

Lignes

174-184	on calcule les moments de chacune des I interférences puis on les combine. Le résultat est dans RES3.	
190	on calcule P_e sans interférence	PE1
192	on calcule P_e en présence des I interférences	PE2

```
0680 --      (*$L+*)
0680 --      PROGRAM QAMBSN(OUTPUT);
0694 --      (*MULTI-INTERFERENCES SINUSOIDALES SANS IIS*)
0694 --      LABEL 5;
0694 --      CONST
0694 --           I=5;
0694 --         FIN=200;
0694 --         PI2=6.283185307179586;
0694 --         LN10=2.302585092994056;
0694 --      TYPE
0694 --         VECTEUR=ARRAY(.1..FIN.) OF REAL;
0694 --      VAR
0694 --         RES1,RES2,RES3:VECTEUR;
1958 --         J,L,N,K,Q:INTEGER;
196C --         CNOB,CNO,SIGMA,SIR:REAL;
1990 --          PE1,PE2:REAL;
19A0 --
19A0 --
19A0 --
19A0 --      (*-------------------------------------------------------------------*)
19A0 --
19A0 --   A  FUNCTION HERMITE(X:REAL;Z:VECTEUR):REAL;
0690 --      VAR
0690 --           J:INTEGER;
0694 --           ARG,TRANS,A,B:REAL;
06B8 --           V:VECTEUR;
0000 0-   A  BEGIN
0018 --           ARG:=(1/SQRT(PI2/2))*EXP(-X*X);
0082 --           V(.1.):=X**3;
00A2 --           V(.2.):=(X**4)*(8*(X**3)-12*X)/24;
0132 --           J:=2;
013A --           TRANS:=V(.1.)*Z(.1.)+V(.2.)*Z(.2.);
017C 1-           REPEAT
017C --                J:=J+1;
018E --                A:=(1-((4*J-5)/(2*X*X)))*V(.J-1.);
0232 --                B:=V(.J-2.)*(J-2)/(J-1);
02A0 --                V(.J.):=(2*(X**4)/(J*(2*J-1)))*(A-B);
0344 --                TRANS:=TRANS+V(.J.)*Z(.J.);
039A -1           UNTIL (ABS(V(.J.)*Z(.J.)/TRANS)<1E-5) OR (J=FIN);
041C --           HERMITE:=TRANS*ARG;
043E --           WRITE('       ',J:3);
046E -0   A  END;
0484 --
0484 --
0484 --      (*-------------------------------------------------------------------*)
0484 --
0484 --   A  FUNCTION COMB(X:REAL;K,N:INTEGER):REAL;
0058 --      (*POUR CALCULER LES COMBINAISONS "K PARMI N".*)
0058 --      VAR
0058 --          I:INTEGER;
0000 0-   A  BEGIN
0018 --         FOR I:=N DOWNTO K+1 DO
004C 1-              BEGIN
004C --                  X:=X*I/(I-K);
0096 -1              END;
00B8 --         COMB:=X;
00CA -0   A  END;
00E8 --
```

```
00E8 --       (*----------------------------------------------------------------------*)
00E8 --
00E8 --  A    PROCEDURE UNION(VAR X,Y,Z:VECTEUR);
004C --       (*POUR COMBINER DEUX MOMENTS.*)
004C --       VAR
004C --           I,J:INTEGER;
0054 --           B,C:REAL;
0000 0-  A    BEGIN
0012 --           FOR I:=1 TO FIN DO Z(.I.):=X(.I.)+Y(.I.);
00C2 --           FOR I:=2 TO FIN DO
00E0 1-                BEGIN
00E0 --                     J:=1;
00E8 2-                     REPEAT
00E8 --                     IF (X(.J.)<>0) AND (Y(.I-J.)<>0) THEN
0188 3-                     BEGIN
0188 --                         B:=LN(Y(.I-J.));
01C8 --                         C:=LN(COMB(X(.J.),2*J,2*I));
0230 --                         IF (B+C)/LN10>-70 THEN
026C --                         Z(.I.):=Z(.I.)+EXP(B+C);
02DC -3                     END;
02DC --                     J:=J+1
02E6 -2                     UNTIL J>=I;
0300 -1                END;
0322 -0  A    END;
034C --
034C --       (*----------------------------------------------------------------------*)
034C --
034C --  A    FUNCTION ERFC(VAR X:REAL):REAL;
004C --       (*CALCUL DE LA FONCTION D'ERREUR COMPLEMENTAIRE PAR LA METHODE
004C --       DE GAUSS D'ORDRE 20*)
004C --
004C --       TYPE
004C --            GAUSS=ARRAY(.1..10.) OF REAL;
004C --       VAR
004C --            T,W:GAUSS;
00F0 --            A,B,R,C:REAL;
0110 --            I:INTEGER;
0114 --
0114 --  B    FUNCTION F(VAR X:REAL):REAL;
004C --       CONST PI=3,14159265358979;
0000 0-  B    BEGIN
0018 --            F:=(2/SQRT(PI))*EXP(-X*X);
0072 -0  B    END;
0094 --
0000 0-  A    BEGIN
0018 --            T(.1.):=0,076526521133497;W(.1.):=0,152753387130725;
0028 --            T(.2.):=0,227785851141645;W(.2.):=0,149172986472603;
0038 --            T(.3.):=0,373706088715419;W(.3.):=0,142096109318382;
0048 --            T(.4.):=0,510867001950827;W(.4.):=0,131688638449176;
0058 --            T(.5.):=0,636053680726515;W(.5.):=0,118194531961518;
0068 --            T(.6.):=0,746331906460150;W(.6.):=0,101930119817240;
0078 --            T(.7.):=0,839116971822218;W(.7.):=0,083276741576704;
0088 --            T(.8.):=0,912234428251325;W(.8.):=0,062672048334109;
0098 --            T(.9.):=0,963971927277913;W(.9.):=0,040601429800386;
00A8 --            T(.10.):=0,993128599185094;W(.10.):=0,017614007139152;
00B8 --            IF ABS(X)>12 THEN R:=0 ELSE
0100 1-                 BEGIN
0100 --                 A:=(13-ABS(X))/2;
```

```
0144 --              B:=(13+ABS(X))/2;
0188 --              R:=0;
019E --              FOR I:=1 TO 10 DO
01BC 2-                   BEGIN
01BC --                   C:=B+A*T(.I.);
0200 --                   R:=R+W(.I.)*F(C);
0252 -2                   END;
0274 --              FOR I:=1 TO 10 DO
0292 2-                   BEGIN
0292 --                   C:=B-A*T(.I.);
02D6 --                   R:=R+W(.I.)*F(C);
0328 -2                   END;
034A -1              END;
034A --         IF X<0 THEN ERFC:=2-A*R ELSE ERFC:=A*R;
03D2 -0 A  END;
0494 --
0494 --    (*-----------------------------------------------------------------------*)
0494 --
0494 -- A  PROCEDURE MOMBS(VAR V:VECTEUR;SIR:REAL;D:INTEGER);
0054 --    (*POUR CALCULER LES MOMENTS NORMALISES D'UN
0054 --      BROUILLEUR SINUSOIDAL. *)
0054 --    VAR
0054 --       I:INTEGER;
0058 --       ARG,S:REAL;
0000 0- A  BEGIN
0012 --       ARG:=10**(-SIR/10);
0064 --         ARG:=ARG*2*(SQR(2*N)-1)/(3*D);
00E6 --       V(.1.):=0.5*ARG;
0100 --        FOR I:=2 TO FIN DO
011E 1-             BEGIN
011E --             IF V(.I-1.)<>0 THEN
0162 2-                  BEGIN
0162 --                  S:=LN(V(.I-1.))+LN(ARG*(2*I-1)/(2*I));
01F4 --                  IF S/LN10>-70 THEN V(.I.):=EXP(S)
0248 --                            ELSE V(.I.):=0;
0288 -2                  END ELSE V(.I.):=0;
02BC -1             END;
02DE --        FOR I:=1 TO FIN DO WRITE(V(.I.):12);
035C --        WRITELN;WRITELN;
0374 -0 A  END;
03AC --
03AC --    (*-----------------------------------------------------------------------*)
03AC --

0000 0-    BEGIN
003A --       CNOB:=18;
0058 --          WRITELN('            ',I:5,'  INTERFERENCES SINUS SANS IIS');
009A --         WRITELN;
00A6 --       WRITELN;WRITELN;
00BE --        FOR N:=2 TO 2 DO
00E8 1-        BEGIN
00E8 --        WRITELN('                   MODULATION QAM=',SQR(2*N):3);
0130 --        WRITELN;WRITELN;
0148 --        FOR L:=0 TO 0 DO
0170 2-        BEGIN
0170 --        SIR:=20-2*L;
01A2 --        WRITELN('      SIR=',SIR:5:1,' DB');
01F8 --        WRITELN;
0204 --        MOMBS(RES2,SIR,I);
```

```
0230 --         FOR Q:=1 TO FIN DO RES3(.Q.):=RES2(.Q.);
02D0 3-         IF I>1 THEN BEGIN
02DC --              FOR K:=1 TO I-1 DO
031E 4-                   BEGIN
031E --                     UNION(RES3,RES2,RES1);
033C --                   FOR Q:=1 TO FIN DO RES3(.Q.):=RES1(.Q.);
03E2 --                   FOR Q:=1 TO FIN DO WRITE(RES3(.Q.):12);
047C --                   WRITELN;WRITELN;
0494 -4                   END;
04CE -3              END;
04CE --        5:FOR J:=0 TO 12 DO
04F6 3-        BEGIN
04F6 --            CNO:=J+CNOB;
0530 --            WRITE('     SNR=',CNO:5:1,' DB');
0580 --            SIGMA:=SQRT(3*(10**(CNO/10))/(2*(SQR(2*N)-1)));
062E --            PE1:=(2-1/N)*0.5*ERFC(SIGMA);
06A8 --            WRITE('       PE1=',PE1:9);
06E4 --            PE2:=(2-1/N)*(0.5*ERFC(SIGMA)+HERMITE(SIGMA,RES3));
07A4 --            WRITELN('    TERMES          PE2=',PE2:9);
07E6 -3         END;
081A -2       END;
084E --         WRITELN;
085A -1     END;
088E -0     END.
```

PROGRAMME QFMRG

Calcule la probabilité d'erreur moyenne due à la sortie d'un récepteur QAM en l'absence d'IIS mais en présence d'une interférence dont la densité spectrale de forme gaussienne est découpée en une infinité de bandes disjointes et indépendantes. On a déjà fait remarquer que cette approche conduit à une interférence de distribution gaussienne.

Lignes

166 calcul de la fraction de la puissance de l'interférence qui entre dans le récepteur.

167 on calcule ici une valeur de SIR après le filtre de réception.

173 on calcule P_e sans interférence.

175 on calcule le rapport signal sur bruit en assimilant l'interférence au bruit gaussien.

178 on calcule P_e en présence de l'interférence de distribution gaussienne.

FC : écart de fréquence entre ls porteuses FM et QAM normalisée à B/2.

DF : écart-type de la densité spectrale FM normalisée à B/2.

```
0680 --      (*SL+*)
0680 --      PROGRAM QFMRG(OUTPUT);
0694 --      (*INTERFERENCE FM DE DENSITE SPECTRALE ET DISTRIBUTION GAUSIENNE*)
0694 --      (*FILTRE IDEAL, SANS IIS ET ROUTINE FLOTTANTE*)
0694 --      CONST
0694 --         PI2=6,283185307179586;
0694 --         LN10=2,302585092994045;
0694 --      VAR
0694 --          R:REAL;
06A0 --         J,L,N,K,Q:INTEGER;
06B4 --         CNOB,CNO,SIGMA,SIR,SIRC,DF,FC:REAL;
06F0 --          PE1,PE2:REAL;
0700 --
0700 --
0700 --      (*------------------------------------------------------------------------*)
0700 --
0700 --   A  FUNCTION BANDE(VAR DF,FC:REAL):REAL;
0050 --      (*CALCUL LA PUISSANCE DANS LA BANDE DU FILTRE*)
0050 --      VAR
0050 --          A,B,C,D,R:REAL;
0078 --          I:INTEGER;
007C --
007C --   B  FUNCTION INT(VAR X,Y:REAL):REAL;
0050 --      (*CALCUL DE L'INTEGRALE D'UNE DENSITE GAUSIENNE ENTRE LES VALEURS X ET Y*)
0050 --
0050 --      TYPE
0050 --          GAUSS=ARRAY(.1..10.) OF REAL;
0050 --      VAR
0050 --          T,W:GAUSS;
00F0 --          A,B,R,C:REAL;
0110 --          I:INTEGER;
0114 --
0114 --   C  FUNCTION FT(VAR X:REAL):REAL;
004C --      CONST PI=3,14159265358979;
0000 0-   C  BEGIN
0018 --          FT:=(1/SQRT(PI))*EXP(-X*X);
0072 -0   C  END;
0094 --
0094 --
0000 0-   B  BEGIN
0018 --          T(.1.):=0,07652652113349;W(.1.):=0,15275338713072;
0028 --          T(.2.):=0,22778585114164;W(.2.):=0,14917298647260;
0038 --          T(.3.):=0,37370608871541;W(.3.):=0,14209610931838;
0048 --          T(.4.):=0,51086700195082;W(.4.):=0,13168863844917;
0058 --          T(.5.):=0,63605368072651;W(.5.):=0,11819453196151;
0068 --          T(.6.):=0,74633190646015;W(.6.):=0,10193011981724;
0078 --          T(.7.):=0,83911697182221;W(.7.):=0,08327674157670;
0088 --          T(.8.):=0,91223442825132;W(.8.):=0,06267204833410;
0098 --          T(.9.):=0,96397192727791;W(.9.):=0,04060142980038;
00A8 --          T(.10.):=0,99312859918509;W(.10.):=0,01761400713915;
00B8 --          IF (ABS(X)>12) OR (ABS(Y)>12) THEN R:=0 ELSE
0142 1-              BEGIN
0142 --              A:=(Y-X)/2;
0182 --              B:=(Y+X)/2;R:=0;
01D8 --              FOR I:=1 TO 10 DO
01F6 2-                  BEGIN
01F6 --                  C:=B+A*T(.I.);
023A --                  R:=R+W(.I.)*FT(C);
```

```
028C -2                      END;
02AE --                FOR I:=1 TO 10 DO
02CC 2-                      BEGIN
02CC --                      C:=B-A*T(,I,);
0310 --                      R:=R+W(,I,)*FT(C);
0362 -2                      END;
0384 -1                END;
0384 --          INT:=A*R;
03A6 -0 B    END;
046C --
0000 0- A    BEGIN
0018 --          R:=0;
002E --          FOR I:=0 TO 9 DO
004A 1-                BEGIN
004A --                A:=-1+(2*I)/10;
0096 --                B:=-1+(2*(I+1))/10;
00E8 --                      C:=(A-FC)/(DF*SQRT(2));
0142 --                      D:=(B-FC)/(DF*SQRT(2));
019C --                      R:=R+INT(C,D);
01CE -1                END;
01F0 --                WRITELN('      ',R:12);
022C --                BANDE:=R;
023E --          WRITELN;WRITELN;
0256 -0 A    END;
0288 --
0288 --      (*---------------------------------------------------------------------*)
0288 --
0288 --
0288 --
0288 -- A    FUNCTION ERFC(VAR X:REAL):REAL;
004C --      (*CALCUL DE LA FONCTION D'ERREUR COMPLEMENTAIRE PAR LA METHODE
004C --      DE GAUSS D'ORDRE 20*)
004C --
004C --      TYPE
004C --          GAUSS=ARRAY(,1,,10,) OF REAL;
004C --      VAR
004C --          T,W:GAUSS;
00F0 --          A,B,R,C:REAL;
0110 --          I:INTEGER;
0114 --
0114 -- B    FUNCTION F(VAR X:REAL):REAL;
004C --      CONST PI=3,14159265358979;
0000 0- B    BEGIN
0018 --          F:=(2/SQRT(PI))*EXP(-X*X);
0072 -0 B    END;
0094 --
0000 0- A    BEGIN
0018 --          T(,1,):=0,076526521133497;W(,1,):=0,152753387130725;
0028 --          T(,2,):=0,227785851141645;W(,2,):=0,149172986472603;
0038 --          T(,3,):=0,373706088715419;W(,3,):=0,142096109318382;
0048 --          T(,4,):=0,510867001950827;W(,4,):=0,131688638449176;
0058 --          T(,5,):=0,636053680726515;W(,5,):=0,118194531961518;
0068 --          T(,6,):=0,746331906460150;W(,6,):=0,101930119817240;
0078 --          T(,7,):=0,839116971822218;W(,7,):=0,083276741576704;
0088 --          T(,8,):=0,912234428251325;W(,8,):=0,062672048334109;
0098 --          T(,9,):=0,963971927277913;W(,9,):=0,040601429800386;
00A8 --          T(,10,):=0,993128599185094;W(,10,):=0,017614007139152;
00B8 --          IF ABS(X)>12 THEN R:=0 ELSE
```

A.41

```
0100 1-              BEGIN
0100 --              A:=(13-ABS(X))/2;
0144 --              B:=(13+ABS(X))/2;
0188 --              R:=0;
019E --              FOR I:=1 TO 10 DO
01BC 2-                   BEGIN
01BC --                   C:=B+A*T(.I.);
0200 --                   R:=R+W(.I.)*F(C);
0252 -2                   END;
0274 --              FOR I:=1 TO 10 DO
0292 2-                   BEGIN
0292 --                   C:=B-A*T(.I.);
02D6 --                   R:=R+W(.I.)*F(C);
0328 -2                   END;
034A -1              END;
034A --         IF X<0 THEN ERFC:=2-A*R ELSE ERFC:=A*R;
03D2 -0 A  END;
0494 --
0494 --    (*-----------------------------------------------------------------------*)
0494 --
0494 --
0000 0-    BEGIN
003A --       CNOB:=18;
0052 --        WRITELN('            INTERFERENCE FM DE DISTRIBUTION GAUSSIENNE');
0070 --        WRITELN;
007C --        WRITELN('            FILTRAGE IDEAL SANS IIS   ');
009A --        WRITELN;
00A6 --        WRITELN('            DENSITE SPECTRALE GAUSSIENNE');
00C4 --         WRITELN;
00D0 --       WRITELN;WRITELN;
00E8 --        FOR N:=2 TO 2 DO
0106 1-        BEGIN
0106 --        WRITELN('                        MODULATION QAM=',SQR(2*N):3);
0148 --        WRITELN;WRITELN;
0160 --        FOR L:=0 TO 0 DO
017C 2-        BEGIN
017C --        SIR:=20-2*L;
01A2 --        WRITELN('        SIR=',SIR:5:1,' DB');
01F2 --        WRITELN;
01FE --        FOR K:=1 TO 5 DO
021C 3-        BEGIN
021C --        DF:=K;
0234 --        WRITELN('        DF=',DF:5:1);
0272 --        WRITELN;
027E --        FOR Q:=0 TO 10 DO
029A 4-        BEGIN
029A --        FC:=Q;
02B2 --        WRITELN('        FC=',FC:5:1);
02F0 --        WRITELN;
02FC --        R:=BANDE(DF,FC);
0318 --        SIRC:=SIR-(10*LN(R)/LN10);
035E --       FOR J:=0 TO 12 DO
037A 5-       BEGIN
037A --           CNO:=J+CNOB;
03A2 --             WRITE('      SNR= ',CNO:5:1,' DB');
03EC --             SIGMA:=SQRT(3*(10**(CNO/10))/(2*(SQR(2*N)-1)));
0488 --             PE1:=(2-1/N)*0.5*ERFC(SIGMA);
04E8 --             WRITE('      PE1= ',PE1:9);
```

```
051E --             CNO:=10*LN(1/((10**(-CNO/10))+(10**(-SIRC/10))))/LN10;
05FA --             WRITE('     SNIR=',CNO:5:1,' DB');
0644 --             SIGMA:=SQRT(3*(10**(CNO/10))/(2*(SQR(2*N)-1)));
06E0 --             PE2:=(2-1/N)*0,5*ERFC(SIGMA);
0740 --             WRITELN('        PE2=',PE2:9);
077C -5          END;
079E --          WRITELN;WRITELN;
07B6 -4          END;
07D8 --          WRITELN;WRITELN;
07F0 -3          END;
0812 --          WRITELN;WRITELN;
082A -2        END;
084C --          WRITELN;WRITELN;
0864 -1     END;
0886 -0     END.
```