



**HAL**  
open science

# Les circuits intégrés de communication de données : architecture et méthodologie de conception

Michel Nguyen-Xuan-Dang

► **To cite this version:**

Michel Nguyen-Xuan-Dang. Les circuits intégrés de communication de données : architecture et méthodologie de conception. Modélisation et simulation. Institut National Polytechnique de Grenoble - INPG, 1988. tel-00331173

**HAL Id: tel-00331173**

**<https://theses.hal.science/tel-00331173>**

Submitted on 15 Oct 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**THÈSE**

présentée à

**l'Institut National Polytechnique de Grenoble**

pour obtenir le grade de  
**DOCTEUR ES-SCIENCES**  
Informatique

par

**Michel Nguyen-Xuan-Dang**

**Les Circuits Intégrés  
de Communication de Données  
Architecture et  
Méthodologie de Conception**

**Thèse soutenue le 15 Décembre 1988 devant la Commission d'Examen :**

**Monsieur J.P. Verjus    Président**

**Messieurs D. Etiemble**

**R. Gerber    Examineurs**

**G. Mazaré**

**G. Michel**



# INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Président : Georges LESPINARD

Année 1988

## Professeurs des Universités

BARIBAUD Michel	ENSERG	JOUBERT Jean-Claude	ENSPG
BARRAUD Alain	ENSIEG	JOURDAIN Geneviève	ENSIEG
BAUDELET Bernard	ENSPG	LACOUME Jean-Louis	ENSIEG
BEAUFILS Jean-Pierre	ENSEEG	LESIEUR Marcel	ENSHMG
BLIMAN Samuel	ENSERG	LESPINARD Georges	ENSHMG
BLOCH Daniel	ENSPG	LONGEQUEUE Jean-Pierre	ENSPG
BOIS Philippe	ENSHMG	LOUCHET François	ENSIEG
BONNETAIN Lucien	ENSEEG	MASSE Philippe	ENSIEG
BOUVARD Maurice	ENSHMG	MASSELOT Christian	ENSIEG
BRISSONNEAU Pierre	ENSIEG	MAZARE Guy	ENSIMAG
BRUNET Yves	IUFA	MOREAU René	ENSHMG
CAILLERIE Denis	ENSHMG	MORET Roger	ENSIEG
CAVAIGNAC Jean-François	ENSPG	MOSSIERE Jacques	ENSIMAG
CHARTIER Germain	ENSPG	OBLED Charles	ENSHMG
CHENEVIER Pierre	ENSERG	OZIL Patrick	ENSEEG
CHERADAME Hervé	UFR PGP	PARIAUD Jean-Charles	ENSEEG
CHOVET Alain	ENSERG	PERRET René	ENSIEG
COHEN Joseph	ENSERG	PERRET Robert	ENSIEG
COUMES André	ENSERG	PIAU Jean-Michel	ENSHMG
DARVE Félix	ENSHMG	POUPOT Christian	ENSERG
DELLA-DORA Jean	ENSIMAG	RAMEAU Jean-Jacques	ENSEEG
DEPORTES Jacques	ENSPG	RENAUD Maurice	UFR PGP
DOLMAZON Jean-Marc	ENSERG	ROBERT André	UFR PGP
DURAND Francis	ENSEEG	ROBERT François	ENSIMAG
DURAND Jean-Louis	ENSIEG	SABONNADIÈRE Jean-Claude	ENSIEG
FOGGIA Albert	ENSIEG	SAUCIER Gabrielle	ENSIMAG
FONLUPT Jean	ENSIMAG	SCHLENKER Claire	ENSPG
FOULARD Claude	ENSIEG	SCHLENKER Michel	ENSPG
GANDINI Alessandro	UFR PGP	SILVY Jacques	UFR PGP
GAUBERT Claude	ENSPG	SIRIEYS Pierre	ENSHMG
GENTIL Pierre	ENSERG	SOHM Jean-Claude	ENSEEG
GREVEN Hélène	IUFA	SOLER Jean-Louis	ENSIMAG
GUERIN Bernard	ENSERG	SOUQUET Jean-Louis	ENSEEG
GUYOT Pierre	ENSEEG	TROMPETTE Philippe	ENSHMG
IVANES Marcel	ENSIEG	VEILLON Gérard	ENSIMAG
JAUSSAUD Pierre	ENSIEG	ZADWORNÝ François	ENSERG

**Professeur Université des Sciences  
Sociales  
( Grenoble II )**

BOLLIET Louis

**Personnes ayant obtenu le diplôme  
d'HABILITATION A DIRIGER DES  
RECHERCHES**

BECKER Monique  
BINDER Zdenek  
CHASSERY Jean-Marc  
CHOLLET Jean-Pierre  
COEY John  
COLINET Catherine  
COMMAULT Christian  
CORNUEJOLS Gérard  
COULOMB Jean- Louis  
DALARD Francis  
DANES Florin  
DEROO Daniel  
DIARD Jean-Paul  
DION Jean-Michel  
DUGARD Luc  
DURAND Madeleine  
DURAND Robert  
GALERIE Alain  
GAUTHIER Jean-Paul  
GENTIL Sylviane  
GHIBAUDO Gérard  
HAMAR Sylvaine  
HAMAR Roger  
LADET Pierre  
LATOMBE Claudine  
LE GORREC Bernard  
MADAR Roland  
MULLER Jean  
NGUYEN TRONG Bernadette  
PASTUREL Alain  
PLA Fernand  
ROUGER Jean  
TCHUENTE Maurice  
VINCENT Henri

**Chercheurs du C.N.R.S  
Directeurs de recherche 1ère Classe**

CARRE René  
FRUCHART Robert  
HOPFINGER Emile  
JORRAND Philippe  
LANDAU Ioan  
VACHAUD Georges  
VERJUS Jean-Pierre

**Directeurs de recherche  
2ème Classe**

ALEMANY Antoine  
ALLIBERT Colette  
ALLIBERT Michel  
ANSARA Ibrahim  
ARMAND Michel  
BERNARD Claude  
BINDER Gilbert  
BONNET Roland  
BORNARD Guy  
CAILLET Marcel  
CALMET Jacques

COURTOIS Bernard  
DAVID René  
DRIOLE Jean  
ESCUDIER Pierre  
EUSTATHOPOULOS Nicolas  
GUELIN Pierre  
JOD Jean-Charles  
KLEITZ Michel  
KOFMAN Walter  
KAMARINOS Georges  
LEJEUNE Gérard  
LE PROVOST Christian  
MADAR Roland  
MERMET Jean  
MICHEL Jean-Marie  
MUNIER Jacques  
PIAU Monique  
SENATEUR Jean-Pierre  
SIFAKIS Joseph  
SIMON Jean-Paul  
SUERY Michel  
TEODOSIU Christian  
VAUCLIN Michel  
WACK Bernard

**Personnalités agréées à titre  
permanent à diriger des travaux de  
recherche (décision du conseil  
scientifique)**

**E.N.S.E.E.G**

CHATILLON Christian  
HAMMOU Abdelkader  
MARTIN GARIN Régina  
SARRAZIN Pierre  
SIMON Jean-Paul

**E.N.S.E.R.G**

BOREL Joseph

**E.N.S.I.E.G**

DESCHIZEAUX Pierre  
GLANGEAUD François  
PERARD Jacques  
REINISCH Raymond

**E.N.S.H.G**

ROWE Alain

**E.N.S.I.M.A.G**

COURTIN Jacques

**E.F.P.**

CHARUEL Robert

**C.E.N.G**

CADET Jean  
COEURE Philippe  
DELHAYE Jean-Marc  
DUPUY Michel  
JOUVE Hubert  
NICOLAU Yvan  
NIFENECKER Hervé  
PERROUD Paul  
PEUZIN Jean-Claude  
TAIB Maurice  
VINCENDON Marc

**Laboratoires extérieurs**

**C.N.E.T**

DEVINE Rodericq  
GERBER Roland  
MERCKEL Gérard  
PAULEAU Yves

# UNIVERSITE Joseph FOURIER (GRENOBLE I)

Président de l'Université :  
M. PAYAN Jean Jacques

Année Universitaire 1987 - 1988

## MEMBRES DU CORPS ENSEIGNANT DE SCIENCES ET DE GEOGRAPHIE

### PROFESSEURS DE 1ère Classe

ARNAUD Paul	Chimie Organique
ARVIEU ROBERT	Physique Nucléaire I.S.N.
AUBERT Guy	Physique C.N.R.S
AURIAULT Jean-Louis	Mécanique
AYANT Yves	Physique Approfondie
BARBIER Marie-Jeanne	Electrochimie
BARJON Robert	Physique Nucléaire ISN
BARNOUD Fernand	Biochimie Macromoléculaire Végétale
BARRA Jean-René	Statistiques-Mathématiques Appliquées
BECKER Pierre	Physique
BEGUIN Claude	Chimie Organique
BELORISKY Elie	Physique
BENZAKEN Claude	Mathématiques Pures
BERARD Pierre	Mathématiques Pures
BERNARD Alain	Mathématiques Pures
BERTRANDIAS Françoise	Mathématiques Pures
BERTRANDIAS Jean-Paul	Mathématiques Pures
BILLET Jean	Géographie
BOELHER Jean-Paul	Mécanique
BONNIER Jane Marie	Chimie Générale
BOUCHEZ Robert	Physique Nucléaire ISN
BRAVARD Yves	Géographie
CARLIER Georges	Biologie Végétale
CAUQUIS Georges	Chimie Organique
CHARDON Michel	Géographie
CHIBON Pierre	Biologie Animale
COHEN ADDAD Jean-Pierre	Physique
COLIN DE VERDIERE Yves	Mathématiques Pures
CYROT Michel	Physique du Solide
DEBELMAS Jacques	Géologie Générale
DEGRANGE Charles	Zoologie
DEMAILLY Jean-Pierre	Mathématiques Pures
DENEUVILLE Alain	Physique
DEPORTES Charles	Chimie Minérale
DOLIQUE Jean-Michel	Physique des Plasmas
DOUCE Roland	Physiologie Végétale
DUCROS Pierre	Cristallographie
FONTAINE Jean-Marc	Mathématiques Pures
GAGNAIRE Didier	Chimie Physique
GERMAIN Jean-Pierre	Mécanique,
GIRAUD Pierre	Géologie
HICTER Pierre	Chimie
IDELMAN Simon	Physiologie Animale
JANIN Bernard	Géographie
JOLY Jean-René	Mathématiques Pures
KAHANE André, détaché	Physique
KAHANE Josette	Physique
KRAKOWIAK Sacha	Mathématiques Appliquées

LAJZEROWICZ Jeanine  
 LAJZEROWICZ Joseph  
 LAURENT Pierre-Jean  
 LEBRETON Alain  
 DE LEIRIS Joël  
 LHOMME Jean  
 LLIBOUTRY Louis  
 LOISEAUX Jean-Marie  
 LUNA Domingo  
 MACHE Régis  
 MASCLE Georges  
 MAYNARD Roger  
 OMONT Alain  
 OZENDA Paul  
 PAYAN Jean-Jacques  
 PEBAY-PEYROULA Jean-Claude  
 PERRIER Guy  
 PIERRARD Jean-Marie  
 PIERRE Jean-Louis  
 RENARD Michel  
 RINAUDO Marguerite  
 ROSSI André  
 SAXOD Raymond  
 SENDEL Philippe  
 SERGERAERT Francis  
 SOUCHIER Bernard  
 SOUTIF Michel  
 STUTZ Pierre  
 TRILLING Laurent  
 VALENTIN Jacques  
 VAN CUTSEM Bernard  
 VIALON Pierre

Physique  
 Physique  
 Mathématiques Appliquées  
 Mathématiques Appliquées  
 Biologie  
 Chimie  
 Géophysique  
 Sciences Nucléaires I.S.N.  
 Mathématiques Pures  
 Physiologie Végétale  
 Géologie  
 Physique du Solide  
 Astrophysique  
 Botanique (Biologie Végétale)  
 Mathématiques Pures  
 Physique  
 Géophysique  
 Mécanique  
 Chimie Organique  
 Thermodynamique  
 Chimie CERMAV  
 Biologie  
 Biologie Animale  
 Biologie Animale  
 Mathématiques Pures  
 Biologie  
 Physique  
 Mécanique  
 Mathématiques Appliquées  
 Physique Nucléaire I.S.N.  
 Mathématiques Appliquées  
 Géologie

#### PROFESSEURS de 2<sup>ème</sup> Classe

ADIBA Michel  
 ANTOINE Pierre  
 ARMAND Gilbert  
 BARET Paul  
 BLANCHI J.Pierre  
 BLUM Jacques  
 BOITET Christian  
 BORNAREL Jean  
 BRUANDET J.François  
 BRUGAL Gérard  
 BRUN Gilbert  
 CASTAING Bernard  
 CERFF Rudiger  
 CHIARAMELLA Yves  
 COURT Jean  
 DUFRESNOY Alain  
 GASPARD François  
 GAUTRON René  
 GENIES Eugène  
 GIDON Maurice  
 GIGNOUX Claude  
 GILLARD Roland  
 GIORNI Alain  
 GONZALEZ SPRINBERG Gérardo  
 GUIGO Maryse  
 GUMUCHAIN Hervé  
 GUITTON Jacques

Mathématiques Pures  
 Géologie  
 Géographie  
 Chimie  
 STAPS  
 Mathématiques Appliquées  
 Mathématiques Appliquées  
 Physique  
 Physique  
 Biologie  
 Biologie  
 Physique  
 Biologie  
 Mathématiques Appliquées  
 Chimie  
 Mathématiques Pures  
 Physique  
 Chimie  
 Chimie  
 Géologie  
 Sciences Nucléaires  
 Mathématiques Pures  
 Sciences Nucléaires  
 Mathématiques Pures  
 Géographie  
 Géographie  
 Chimie

HACQUES Gérard  
 HERBIN Jacky  
 HERAULT Jeanny  
 JARDON Pierre  
 JOSELEAU Jean-Paul  
 KERCKHOVE Claude  
 LONGEQUEUE Nicole  
 LUCAS Robert  
 MANDARON Paul  
 MARTINEZ Francis  
 NEMOZ Alain  
 OUDET Bruno  
 PECHER Arnaud  
 PELMONT Jean  
 PERRIN Claude  
 PFISTER Jean-Claude  
 PIBOULE Michel  
 RAYNAUD Hervé  
 RICHARD Jean-Marc  
 RIEDTMANN Christine  
 ROBERT Gilles  
 ROBERT Jean-Bernard  
 SARROT-REYNAULD Jean  
 SAYETAT Françoise  
 SERVE Denis  
 STOECKEL Frédéric  
 SCHOLL Pierre-Claude  
 SUBRA Robert  
 VALLADE Marcel  
 VIDAL Michel  
 VIVIAN Robert  
 VOTTERO Philippe

Mathématiques Appliquées  
 Géographie  
 Physique  
 Chimie  
 Biochimie  
 Géologie  
 Sciences Nucléaires I.S.N.  
 Physique  
 Biologie  
 Mathématiques Appliquées  
 Thermodynamique CNRS - CRTBT  
 Mathématiques Appliquées  
 Géologie  
 Biochimie  
 Sciences Nucléaires I.S.N.  
 Physique du Solide  
 Géologie  
 Mathématiques Appliquées  
 Physique  
 Mathématiques Pures  
 Mathématiques Pures  
 Chimie Physique  
 Géologie  
 Physique  
 Chimie  
 Physique  
 Mathématiques Appliquées  
 Chimie  
 Physique  
 Chimie Organique  
 Géographie  
 Chimie

## MEMBRES DU CORPS ENSEIGNANT DE L' IUT 1

### PROFESSEURS de 1<sup>ère</sup> Classe

BUISSON Roger  
 DODU Jacques  
 NEGRE Robert  
 NOUGARET Marcel  
 PERARD Jacques

Physique IUT 1  
 Mécanique Appliquée IUT 1  
 Génie Civil IUT 1  
 Automatique IUT 1  
 EEA. IUT 1

### PROFESSEURS de 2<sup>ème</sup> classe

BOUTHINON Michel  
 CHAMBON René  
 CHEHIKIAN Alain  
 CHENAVAS Jean  
 CHOUTEAU Gérard  
 CONTE René  
 GOSSE Jean-Pierre  
 GROS Yves  
 KUHN Gérard, (Détaché)  
 MAZUER Jean  
 MICHOUlier Jean  
 MONLLOR Christian  
 PEFFEN René  
 PERRAUD Robert  
 PIERRE Gérard  
 TERRIEZ Jean-Michel  
 TOUZAIN Philippe  
 VINCENDON Marc

EEA. IUT 1  
 Génie Mécanique IUT 1  
 EEA. IUT 1  
 Physique IUT 1  
 Physique IUT 1  
 Physique IUT 1  
 EEA.IUT 1  
 Physique IUT 1  
 Physique IUT 1  
 Physique IUT 1  
 Physique IUT 1  
 EEA.IUT 1  
 Métallurgie IUT 1  
 Chimie IUT 1  
 Chimie IUT 1  
 Génie Mécanique IUT 1  
 Chimie IUT 1  
 Chimie IUT 1



## PROFESSEURS DE PHARMACIE

AGNIUS-DELORD Claudine	Physique	Faculté La Tronche
ALARY Josette	Chimie Analytique	Faculté La Tronche
BERIEL Hélène	Physiologie et Pharmacologie	Faculté La Tronche
CUSSAC Max	Chimie Thérapeutique	Faculté La Tronche
DEMENGE Pierre	Pharmacodynamie	Faculté La Tronche
FAVIER Alain	Biochimie	C.H.R.G.
JEANNIN Charles	Pharmacie Galénique	Faculté Meylan
LATURAZE Jean	Biochimie	Faculté La Tronche
LUU DUC Cuong	Chimie Générale	Faculté La Tronche
MARIOTTE Anne-Marie	Pharmacognosie	Faculté La Tronche
MARZIN Daniel	Toxicologie	Faculté Meylan
RENAUDET Jacqueline	Bactériologie	Faculté La Tronche
ROCHAT Jacques	Hygiène et Hydrologie	Faculté La Tronche
SEIGLE-MURANDI Françoise	Botanique et Cryptogamie	Faculté Meylan
VERAIN Alice	Pharmacie Galénique	Faculté Meylan

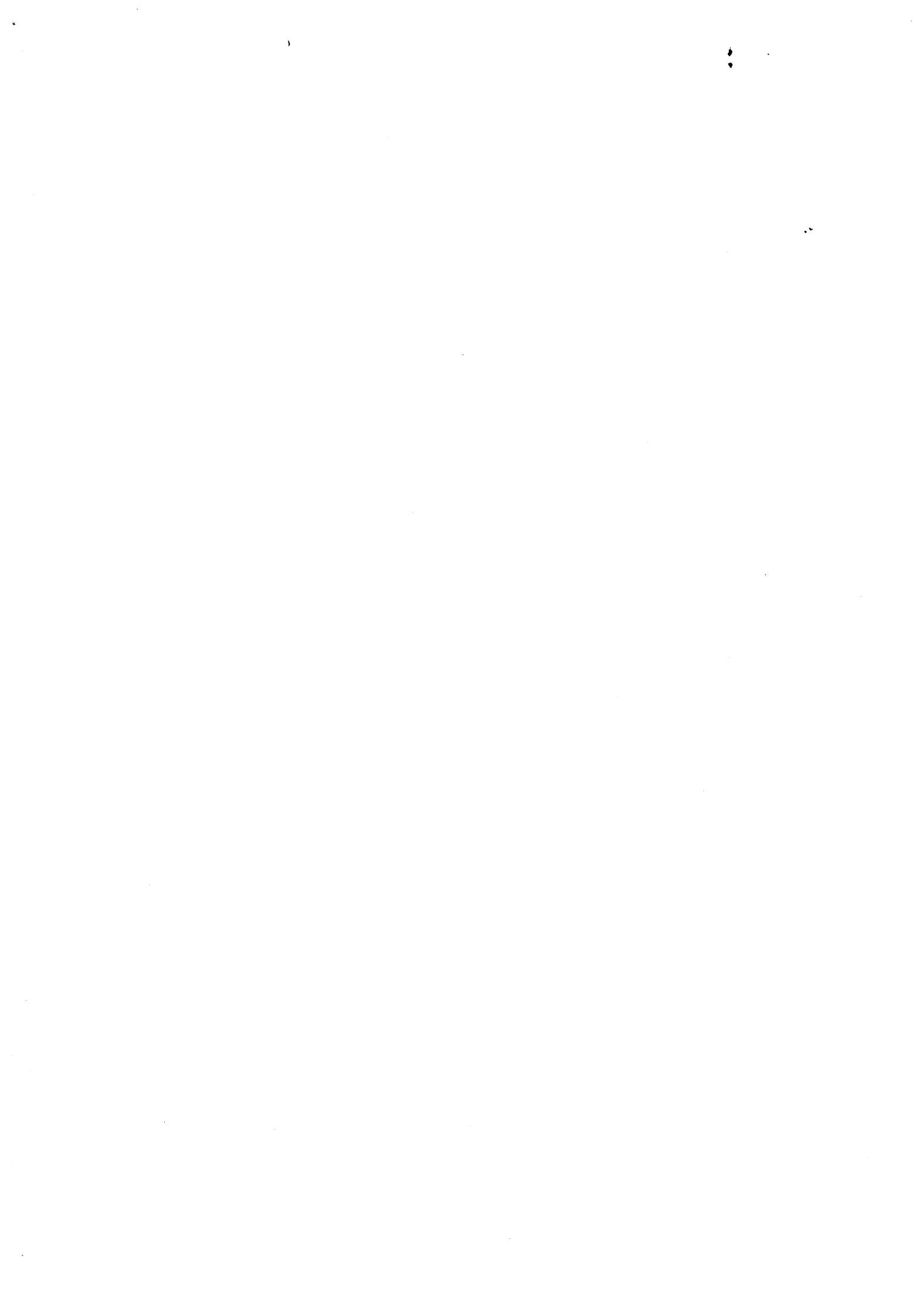
## MEMBRES DU CORPS ENSEIGNANT DE MEDECINE

### PROFESSEURS CLASSE EXEPTIONNELLE ET 1ère CLASSE

AMBLARD Pierre	Dermatologie	C.H.R.G.
AMBROISE-THOMAS Pierre	Parasitologie	C.H.R.G.
BEAUDOING André	Pédiatrie-Puericulture	C.H.R.G.
BEZEZ Henri	Orthopédie-Traumatologie	Hopital SUD
BONNET Jean-Louis	Ophthalmologie	C.H.R.G.
BOUCHET Yves	Anatomie	Faculté La Merci
	Chirurgie Générale et Digestive	C.H.R.G.
BUTEL Jean	Orthopédie-Traumatologie	C.H.R.G.
CHAMBAZ Edmond	Biochimie	C.H.R.G.
CHAMPETIER Jean	Anatomie-Topographique	
	et Appliquée	C.H.R.G.
CHARACHON Robert	O.R.L.	C.H.R.G.
COLOMB Maurice	Immunologie	Hopital sud
COUDERC Pierre	Anatomie-Pathologique	C.H.R.G.
DELORMAS Pierre	Pneumophtisiologie	C.H.R.G.
DENIS Bernard	Cardiologie	C.H.R.G.
GAVEND Michel	Pharmacologie	Faculté La Merci
HOLLARD Daniel	Hématologie	C.H.R.G.
LATREILLE René	Chirurgie Thoracique et	
	Cardiovasculaire	C.H.R.G.
LE NOC Pierre	Bactériologie-Virologie	C.H.R.G.
MALINAS Yves	Gynécologie et Obstétrique	C.H.R.G.
MALLION Jean-Michel	Médecine du Travail	C.H.R.G.
MICOUD Max	Clinique Médicale et Maladies	
	Infectieuses	C.H.R.G.
MOURIQUAND Claude	Histologie	Faculté La Merci
PARAMELLE Bernard	Pneumologie	C.H.R.G.
PERRET Jean	Neurologie	C.H.R.G.
RACHAIL Michel	Hépto-Gastro-Entérologie	C.H.R.G.
DE ROUGEMONT Jacques	Neurochirurgie	C.H.R.G.
SARRAZIN Roger	Clinique Chirurgicale	C.H.R.G.
STIEGLITZ Paul	Anestésiologie	C.H.R.G.
TANCHE Maurice	Physiologie	Faculté La Merci
VIGNAIS Pierre	Biochimie	Faculté La Merci

**PROFESSEURS 2ème CLASSE**

BACHELOT Yvan	Endocrinologie	C.H.R.G.
BARGE Michel	Neurochirurgie	C.H.R.G.
BENABID Alim Louis	Biophysique	Faculté La Merci
BENSA Jean-Claude	Immunologie	Hopital Sud
BERNARD Pierre	Gynécologie-Obstétrique	C.H.R.G.
BESSARD Germain	Pharmacologie	ABIDJAN
BOLLA Michel	Radiothérapie	C.H.R.G.
BOST Michel	Pédiatrie	C.H.R.G.
BOUCHARLAT Jacques	Psychiatrie Adultes	Hopital Sud
BRAMBILLA Christian	Pneumologie	C.H.R.G.
CHIROSEL Jean-Paul	Anatomie-Neurochirurgie	C.H.R.G.
COMET Michel	Biophysique	Faculté La Merci
CONTAMIN Charles	Chirurgie Thoracique et Cardiovasculaire	C.H.R.G.
CORDONNIER Daniel	Néphrologie	C.H.R.G.
COULOMB Max	Radiologie	C.H.R.G.
CROUZET Guy	Radiologie	C.H.R.G.
DEBRU Jean-Luc	Médecine Interne et Toxicologie	C.H.R.G.
DEMONGEOT Jacques	Biostatistiques et Informatique Médicale	Faculté La Merci
DUPRE Alain	Chirurgie Générale	C.H.R.G.
DYON Jean-François	Chirurgie Infantile	C.H.R.G.
ETERRADOSSI Jacqueline	Physiologie	Faculté La Merci
FAURE Claude	Anatomie et Organogénèse	C.H.R.G.
FAURE Gilbert	Urologie	C.H.R.G.
FOURNET Jacques	Hépatogastro-Entérologie	C.H.R.G.
FRANCO Alain	Médecine Interne	C.H.R.G.
GIRARDET Pierre	Anesthésiologie	C.H.R.G.
GUIDICELLI Henri	Chirurgie Générale et Vasculaire	C.H.R.G.
GUIGNIER Michel	Thérapeutique et Réanimation Médicale	C.H.R.G.
HADJIAN Arthur	Biochimie	Faculté La Merci
HALIMI Serge	Endocrinologie et Maladies Métaboliques	C.H.R.G.
HOSTEIN Jean	Hépatogastro-Entérologie	C.H.R.G.
HUGONOT Robert	Médecine Interne	C.H.R.G.
JALBERT Pierre	Histologie-Cytogénétique	C.H.R.G.
JUNIEN-LAVILLAULOY Claude	O.R.L.	C.H.R.G.
KOLODIE Lucien	Hématologie Biologique	C.H.R.G.
LETOUBLON Christian	Chirurgie Générale	C.H.R.G.
MACHECOURT Jacques	Cardiologie et Maladies Vasculaires	C.H.R.G.
MAGNIN Robert	Hygiène	C.H.R.G.
MASSOT Christian	Médecine Interne	C.H.R.G.
MOUILLON Michel	Ophthalmologie	C.H.R.G.
PELLAT Jacques	Neurologie	C.H.R.G.
PHELIP Xavier	Rhumatologie	C.H.R.G.
RACINET Claude	Gynécologie-Obstétrique	Hopital Sud
RAMBAUD Pierre	Pédiatrie	C.H.R.G.
RAPHAEL Bernard	Stomatologie	C.H.R.G.
SCHAERER René	Cancérologie	C.H.R.G.
SEIGNEURIN Jean-Marie	Bactériologie-Virologie	Faculté La Merci
SELE Bernard	Cytogénétique	Faculté La Merci
SOTTO Jean-Jacques	Hématologie	C.H.R.G.
STOEBNER Pierre	Anatomie Pathologique	C.H.R.G.
VROUSOS Constantin	Radiothérapie	C.H.R.G.



à ma Mère

Je ne regarderai ni l'or du soir qui tombe  
Ni les voiles au loin descendant vers Harfleur  
Victor Hugo



à Annie

à Guillaume, Jean-Marie et Yoko



La première pensée est pour Jean du Masle qui nous a quittés au printemps 1976. Il avait guidé mes premiers travaux sur les réseaux d'ordinateurs.

Les remerciements, multiples et respectueux, vont d'abord à tous ceux qui, d'une manière ou d'une autre, ont orienté ou contribué à mes activités de recherche.

Mr le Professeur Jean-Pierre Verjus, de l'Institut National Polytechnique de Grenoble, qui a dirigé ma thèse de troisième cycle sur les systèmes distribués en 1978 et qui me fait le grand honneur de présider ce jury de thèse.

Mr le Professeur Guy Mazaré, de l'ENSIMAG, qui m'a accueilli dans son équipe en 1982 et qui m'a constamment encouragé dans la voie choisie, avec le tact et le très fin esprit d'analyse et de synthèse qui le caractérisent.

Mr Gérard Michel, actuellement Directeur Technique et des Etudes Avancées chez APTOR S.A., avec lequel j'ai commencé les travaux sur les circuits de communication alors qu'il était Chef de Département au CNET CNS. Il a toujours su faire la part de ce qui est original dans les recherches appliquées que nous menons. Les échanges fructueux que nous avons toujours eus ensemble ont sans conteste contribué à la rédaction de cette thèse.

Mr le Professeur Louis Bolliet qui m'a témoigné depuis que j'avais intégré l'équipe du Professeur André Laplace, sollicitude, confiance et encouragement.

Mr le Professeur Roland Gerber, Chef de Division au CNET CNS, qui m'a fait le redoutable honneur de juger ce travail. La collaboration CNET CNS - IMAG LGI doit beaucoup à sa clairvoyance et à son jugement.

Je tiens à remercier Mr le Professeur Daniel Etiemble, de l'Université de Paris Orsay, de l'intérêt qu'il porte à mes travaux en acceptant d'en être rapporteur et Mr Denis Rouquier, Chef du Département AMS au CNET CNS, qui a toujours su me faire partager son sens de la rigueur dans l'analyse des problèmes.



Une thèse d'Etat est aussi un travail d'équipe. Mes remerciements les plus chaleureux vont à Messieurs Dante Couto-Barone, Mario Diaz-Nava, Christophe Diot, Philippe Objois, Imad Sabouni et Luc Sponga pour les travaux en commun que nous avons effectués. Je ne voudrais surtout pas oublier les différents collègues et partenaires du CNET CNS, de APTOR S.A. et de DOLPHIN S.A., avec lesquels nous avons eu des discussions constructives.

En tant qu'enseignant au département d'IUT GEA de Valence, je tiens à remercier un ami de longue date, Mr Antoine de Lacheisserie, Directeur-Adjoint de l'IUT 2 de Grenoble chargé du Centre de Valence, et l'ensemble des personnes qui ont fait tourner l'institut et qui m'ont fait leur, alors que j'étais et je suis quelque part dans un laboratoire de recherche.

Je tiens à remercier la correctrice des épreuves, puriste de la langue en regard d'une écriture mal achevée, qui m'a appris le nous conventionnel et le subjonctif lexicalisé, Madame Madeleine Rambaud.

Un dernier mot pour un homme très connu et méconnu, un scientifique rigoureux et austère, un ingénieur de vocation, un ami : Michel Delaunay.

# LES CIRCUITS INTEGRES VLSI DE COMMUNICATION DE DONNEES ARCHITECTURE ET METHODOLOGIE DE CONCEPTION

## Avant-Propos

### 1. Introduction

- 1.1. Nécessité de l'intégration
- 1.2. Quels circuits de communication ?
- 1.3. Problématique du domaine de recherche : Les circuits de communication ASIC, des spécifications à la réalisation
- 1.4. Domaines de recherche connexes

### 2. Architecture des circuits de communication pour les couches 2 ISO ou équivalentes

- 2.1. Introduction
- 2.2. Voie Parallèle
  - 2.21. Architecture fonctionnelle des cartes coupleur de connexion aux bus multiprocesseurs
  - 2.22. Travaux réalisés
    - 2.221. Etude de l'architecture et Conception d'un circuit arbitre de bus multi-protocoles : ABCM
    - 2.222. Etude de l'architecture et Conception de la carte coupleur de connexion de micro-ordinateurs hétérogènes à un bus à contrôle distribué
  - 2.23. Recommandations pour les choix d'architecture matérielle
- 2.3. Voie Série
  - 2.31. Introduction aux réseaux locaux
  - 2.32. Architecture fonctionnelle des cartes coupleur et des circuits contrôleur de communication

**2.33. Travaux réalisés**

2.331. Etude de l'architecture du circuit FIP-VLSI

2.332. Etude de l'architecture d'un circuit CSMA/DCR

2.34. Recommandations pour les choix d'architecture matérielle

**3. Architecture de circuit de communication pour les couches 3, 4, 5 ISO ou équivalentes**

3.1. Introduction

3.2. Architecture fonctionnelle du circuit de communication MC3

3.21. Analyse de synthèse des implantations logicielles et matérielles existant

3.22. Architecture fonctionnelle cible

3.23. Architectures des systèmes utilisant MC3

3.24. Complexité des structures de données

3.25. Architecture des machines de transfert

3.3. Vers un microprocesseur de communication

**4. Méthodologie de conception des circuits de communication**

4.1. Présentation de la méthodologie

4.2. Conception des circuits de communication : travaux réalisés

4.21. Conception du circuit FIP-VLSI

4.22. Définition et Validation de l'architecture de MC3

4.3. Etude d'un compilateur de blocs flexibles DMA

**5. Conclusions**

**Références**

**Les Circuits Intégrés  
de Communication de Données  
Architecture et  
Méthodologie de Conception**

**Michel Nguyen-Xuan-Dang**

## AVANT-PROPOS

De 1975 à 1981, les sujets de mes activités de recherche étaient dans le domaine des systèmes répartis, domaine dans lequel j'ai situé ma thèse de 3ème cycle : "Système et Langage Portable pour le Traitement des Applications Réparties".

Depuis le début des années 1980, le domaine de la communication digitale a pris beaucoup d'ampleur. Différents domaines de recherche ont été explorés [INR 83], les résultats publiés. Citons quelques uns de ces domaines : ils vont des systèmes répartis à la spécification et la description des protocoles de communication, de l'évaluation de performances à l'étude théorique des algorithmes de gestion d'accès aux réseaux locaux, etc ... Dans ces différents domaines, les résultats des étapes de recherche expérimentale conduisent déjà aux tentatives de formalisation et structuration théoriques.

Parallèlement au développement de la communication digitale, le domaine de l'architecture des ordinateurs a connu aussi une mutation profonde, que ce soit à l'intérieur des architectures de type Von Neumann (les architectures RISC par exemple) ou en allant vers des architectures moins conventionnelles (les architectures à parallélisme massif par exemple) [LaR 88].

L'influence des progrès technologiques se faisant sentir chaque jour de manière plus convaincante que la veille, le domaine de *l'implantation matérielle des protocoles de communication* m'a semblé devoir être exploré, aussi bien sous *l'aspect des architectures fonctionnelles des machines dédiées à la communication digitale* que sous *l'aspect des méthodes et outils de conception* pour réaliser lesdites machines sous forme de circuits intégrés. L'exposé qui suit reflète la démarche expérimentale qui nous a guidé.

Démarche d'exploration d'un domaine, mais démarche personnelle aussi de quelqu'un dont la culture de base était essentiellement logicielle et qui avait envie un jour de voir de plus près les circuits intégrés à haute densité d'intégration dédiés à la communication.

## **Chapitre 1**

### **INTRODUCTION**

Dans le domaine des systèmes de communication de données, en particulier dans celui des réseaux locaux et des télécommunications, nous assistons actuellement à une grande diversité dans les annonces des produits et à l'émergence de nouveaux produits [LeL 84], [HAR 85], [TOB 86], [AOY 87]. Une préoccupation commune partagée par les constructeurs de ces réseaux, par les différentes compagnies de téléphone privées ou publiques et par les constructeurs d'ordinateurs, se situe au niveau :

- . des circuits intégrés gérant les différents protocoles et les différentes méthodes d'accès, nommés ci-après contrôleurs de communication ou plus généralement circuits de communication, ainsi que
- . du choix des interfaces avec le système hôte (ou station).

### 1.1. NECESSITE DE L'INTEGRATION

Dans le domaine de la communication, comme dans des secteurs connexes (traitement du signal, traitement d'images), l'évolution technique et l'expansion des besoins ont été importantes pendant ces dix dernières années. Elles ont été supportées par la large utilisation de la technologie LSI. Cependant, il est apparent aujourd'hui que dans le domaine de la communication locale ou dans celui des télécommunications, les changements en terme qualitatif et en terme quantitatif vont s'opérer sur une échelle encore plus grande.

Nécessité technique d'abord. Le traitement analogique fait place petit à petit au traitement digital. Le traitement digital est conduit à une intégration de plus en plus poussée. Du point de vue technique, le moteur de ces changements provient en large part des percées technologiques effectuées en VLSI. Dans ce cadre, les circuits intégrés dédiés (ASIC) à très haute densité d'intégration constituent un objectif de première importance. Et par voie de conséquence, l'étude et le prototypage de ce type de circuits deviennent des sujets de recherche d'importance grandissante.

Les problèmes techniques sont généralement liés :

- . aux contraintes de vitesse, de débit, de consommation électrique,
- . aux contraintes d'interfaces et de facilité d'utilisation (modularité, portabilité, observabilité dynamique) et
- . aux contraintes de qualité.

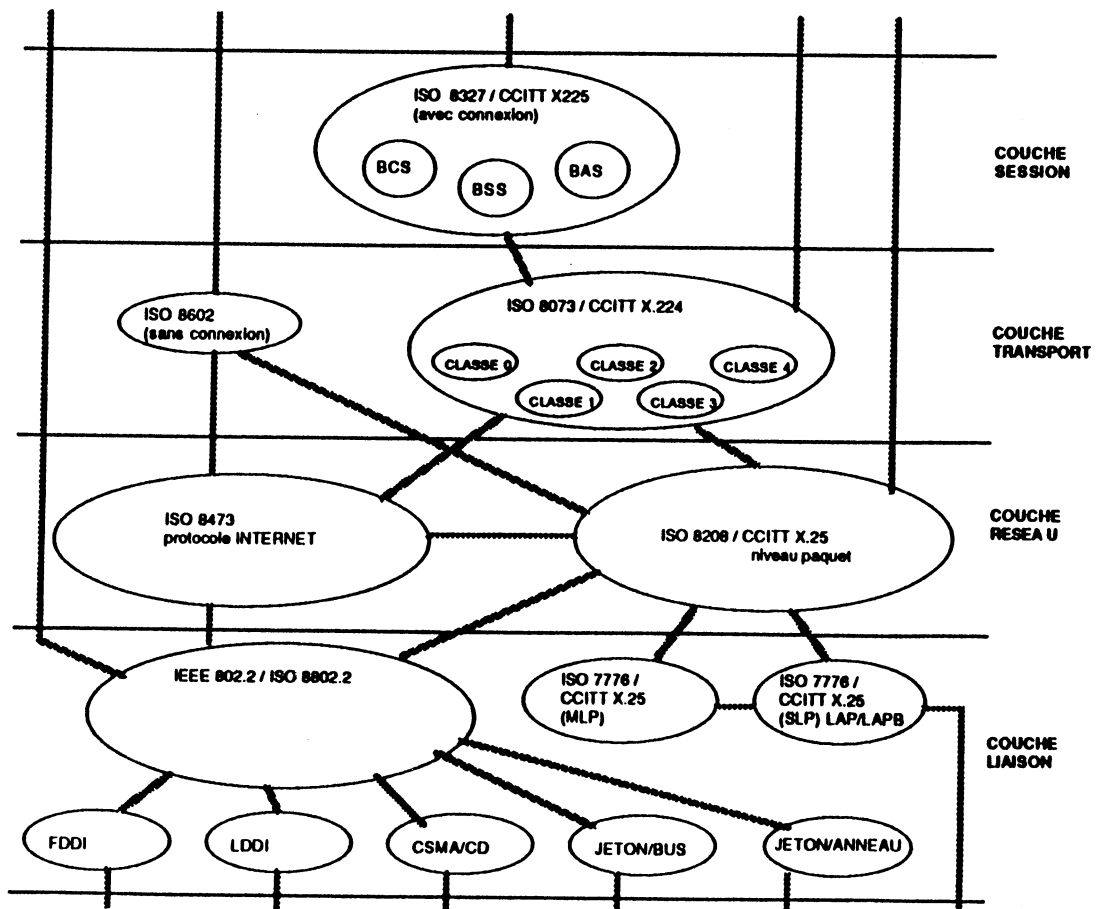
Ces contraintes sont souvent contradictoires et impliquent des solutions de compromis dans les choix de spécification. Dans le domaine des circuits de communication, le critère prédominant reste généralement le critère de vitesse et débit.

D'autre part, la démarche conduisant à l'intégration répond à des nécessités industrielles telles que le besoin de figer des concepts pour obliger le marché à converger, la notion de secret industriel et la recherche de standards de fait. Les contraintes industrielles d'environnement et de taille peuvent entraîner aussi à l'obligation de réaliser un circuit de communication (et non une carte coupleur de communication faite à partir de composants standards et de composants programmables) pour connecter une station à un réseau [DAN 86.4].

L'intégration permet, en dernier lieu, la prise en compte de fonctionnalités additionnelles souvent délaissées dans la réalisation des cartes, telles les fonctionnalités de test par exemple.

## 1.2. QUELS CIRCUITS DE COMMUNICATION ?

Répondre à cette question revient à répondre à la question concernant le domaine que nous nous sommes proposé d'explorer. Prenons le modèle de référence OSI de l'ISO pour permettre de découper fonctionnellement les différents besoins de communication. Nous nous sommes intéressé aux couches intermédiaires, *de la couche Session à la couche Liaison*. En face de chaque niveau de besoins, se trouvent cités les différents standards et normes bâtis pour essayer de les satisfaire.



Les couches 2, 3, 4, 5 ISO et CCITT. (Copyright RETIX™)



. La couche Session permet à deux activités informatiques distantes de communiquer; le terme 'activité' est pris au sens large de programme.

. La couche Transport permet à deux processus informatiques distants de communiquer; le terme processus est pris dans le sens où un ou plusieurs processus concourent à la réalisation d'une activité.

. La couche Réseau permet à deux nœuds distants d'un maillage de connexion donné de communiquer; le terme nœud est pris dans le sens de processeur physique. Ce maillage peut être, celui d'un réseau général ou celui de la connexion d'un réseau local à un réseau général ou celui de la connexion de deux ou de plusieurs réseaux locaux.

. La couche Liaison se décompose généralement en deux sous-couches, si l'on suit le découpage proposé par le standard IEEE 802 [IEE 84] :

+ La sous-couche de Contrôle de Liaison Logique (appelé LLC) : elle permet de contrôler logiquement la connexion de deux processeurs physiques soit en point-à-point soit en multipoints.

+ La sous-couche de Contrôle de la Méthode d'Accès (appelé MAC) : elle permet de contrôler physiquement cette connexion; son rôle est particulièrement manifeste dans le cas des réseaux locaux. L'interface avec la couche Physique est traitée à ce niveau.

Les circuits intégrés gérant les connexions physiques au medium ne font pas l'objet de l'exposé suivant, dans la mesure où les problèmes liés aux technologies propres de ce type de circuits (différentes technologies MOS spécifiques, technologie bipolaire, biMOS) sont de nature plus analogique que digitale.

Le plan du développement qui suit consacre un premier chapitre (§ 2) aux circuits de communication qui traitent les fonctionnalités de la couche MAC avec éventuellement certaines fonctionnalités de la couche LLC (pour des protocoles qui ne respectent pas le découpage ci-dessus). La réflexion sur les architectures des circuits de communications ne peut pas être menée indépendamment des interfaces avec les circuits microprocesseurs. De ce fait, le sous-chapitre 2.2 est consacré aux coupleurs de connexion des bus parallèles multiprocesseurs. La frontière fonctionnelle entre un bus parallèle multiprocesseurs et un bus série réseau local est illustrée par le circuit de communication du bus SLAN (§ 2.222).

Le deuxième chapitre sur les circuits de communication (§ 3) porte sur les couches hautes, allant de la couche LLC à la couche Session. L'étude de l'architecture des circuits de ces couches ainsi que les propositions correspondantes font appel aux résultats obtenus dans le chapitre précédent. Dans l'architecture du processeur MC3 proposé :

- . l'architecture interne de MC3 utilise des bus multiprocesseurs,
- . les environnements système de MC3 utilisent des bus fond-de-panier,
- . la machine de transfert de MC3 est une machine dont l'architecture cible est celle définie dans le chapitre précédent.

Le dernier chapitre (§ 4) porte sur la méthodologie de conception des circuits de communication et décrit principalement nos expériences menées dans le cadre de la conception du circuit FIP-VLSI et de la validation de l'architecture du circuit MC3.

### **1.3. PROBLEMATIQUE DU DOMAINE DE RECHERCHE : LES CIRCUITS DE COMMUNICATION ASIC, DES SPECIFICATIONS A LA REALISATION**

Le fait de vouloir étudier les circuits intégrés de communication à l'Université nous amène à poser et à résoudre les problèmes suivants :

#### **1.31. L'étude des circuits commence par l'étude de leurs architectures et de leurs interfaces.**

L'étude d'architecture que nous avons menée est un type d'étude que nous qualifions d'**expérimentale** [AGU 87], dans la mesure où la démarche adoptée consiste à analyser l'existant, proposer une solution technologiquement viable et à jour, la valider ou la réaliser, puis soumettre les résultats à l'observation et à la critique.

Les grandes lignes de notre approche architecturale se situent dans le domaine des architectures multiprocesseurs. Cependant, les deux approches internes de ce domaine cohabitent, comme nous allons le voir : l'aspect multiprocesseurs câblés (ou multi-unités asynchrones câblées) pour une classe donnée de circuits de communication et l'aspect processeur à usage multiple (et à architecture parallèle) pour l'autre classe.

#### **A. ARCHITECTURE FONCTIONNELLE**

L'étude de l'architecture fonctionnelle d'un circuit de communication doit être précédée d'une étude système qui débouche sur des spécifications du circuit. Cette étude système doit tenir compte de l'existant, faire l'investigation des services que le circuit doit offrir, définir les choix d'interfaces et prévoir les extensions possibles au circuit.

L'architecture fonctionnelle dépend de la complexité du protocole et des interfaces que le circuit est censé implanter.

Dans le domaine couvert par la couche MAC du standard IEEE ou dans des domaines équivalents, faire l'état de l'art concernant les architectures des différents circuits intégrés existants, leurs interfaces et développer le savoir-faire correspondant en conception ont nécessité un effort de longue durée. Nous avons pu faire la synthèse des différents types d'architecture et des propositions sur les choix d'architecture fonctionnelle cible (§ 2).

Le deuxième domaine ne fournit pour ainsi dire pas d'exemple (académique ou industriel) d'implantation sous forme de circuit intégré; par conséquent, il représente un creuset pour des sujets de recherche actuels et à venir. Notre contribution d'architecture fonctionnelle prend la forme d'un *processeur dédié à usage général* dont l'architecture à parallélisme asynchrone est relativement complexe (évalué à plus de 750.000 transistors), architecture dont la validation vient d'être menée à bien (§ 3).

## B. ETUDE DE FAISABILITE TECHNOLOGIQUE

L'étude de la faisabilité technologique d'un circuit dans une technologie donnée précède la conception même du circuit. Elle nécessite une étude de la complexité globale (facteurs de régularité, complexité spécifique des modules et des opérateurs exprimée en unités mesurables, ...), une estimation précise de la surface du futur circuit, une analyse des performances temporelles (débit, temps de réponse), électriques et autres (thermiques, mécaniques, ...). Cette étude de faisabilité amène généralement des retours arrière dans l'étude d'architecture fonctionnelle citée ci-dessus.

### 1.22. Pourquoi étudier à l'Université les circuits intégrés à la demande et dédiés à un domaine donné, ainsi que les méthodes et outils associés ?

Y-a-t-il une demande de ces méthodes et outils ? La réponse est oui. Il suffit de se reporter aux difficultés, résolues depuis, mais connues du grand public de la mise au point des différents circuits ETHERNET (niveau MAC de l'IEEE 802), de se reporter aux difficultés actuelles ne serait-ce que dans la mise au point des circuits pour les protocoles X25, sans parler des futurs circuits intégrés pour de nouveaux besoins de communication locale, ceux en prévision du marché RNIS [AOY 87], et ceux traitant les couches hautes de l'ISO ou équivalents (CCITT [CCI 84], MAP [KAM 86] [MAP 84] CNMA [ESP 86]).

Il ne s'agit pas, bien entendu, pour une équipe de recherches comme la nôtre de réaliser des circuits intégrés à la demande, sinon des circuits prototype. Nous avons choisi par contre d'adopter une **méthodologie de conception** pour aider des ingénieurs système à réaliser rapidement et sûrement des VLSI prototypes, spécialisés dans le domaine de la communication.

Cette étude nous a conduit à spécifier et à développer des outils tels que *bibliothèque de cellules*, *bibliothèque d'opérateurs flexibles spécialisés* et *bibliothèques de modules paramétrables spécialisés* pour la conception de ces circuits de communication.

Un des buts à terme est de produire des *compilateurs de silicium* de plus en plus performants ayant pour objectif la *synthèse automatique des circuits intégrés*, dédiés dans notre cas à la communication.

## **1.4. DOMAINES DE RECHERCHE AVEC LE MEME TYPE D'OBJECTIFS OU A PROBLEMATIQUE EQUIVALENTE**

### **1.4.1. Domaine des circuits de communication**

Le domaine de recherche dans lequel nous nous situons ainsi que notre problématique a une forte connotation de Recherche et Développement. Il n'est donc pas étonnant que dans les citations suivantes, une place de choix soit réservée aux recherches menées par des industriels.

Citons dans l'ordre chronologique d'aboutissement les projets suivants :

. Le projet d'un ensemble évolutif de modules VLSI pour le réseau RNIS, mené par ALCATEL BELL Telephone [RAB 88] : Il s'agit d'une approche flexible de conception de modules VLSI pour introduire les fonctionnalités de RNIS dans un système propriétaire donné. L'ensemble comprend trois modules de transmission pour l'accès primaire, un module paramétrable pour les fonctionnalités de niveau 2, un module d'interface avec le réseau et un module d'interface adaptative pour la station application. L'approche suivie par ce projet, aussi bien en terme de partition fonctionnelle qu'en terme de méthode de conception (outils de CAO et contraintes technologiques), est voisine de notre approche.

. Le projet des macro-cellules dédiées aux fonctions de télécommunication mené par MHS. Il s'agit de la conception de trois macro-cellules paramétrables. L'ensemble comprend un module émetteur-receveur synchrone en "full-duplex", un module qui gère des squelettes de trame HDLC et un module DMA. L'approche suivie par ce projet fait que les macro-cellules ne sont pas vraiment flexibles dans la mesure où elles concernent des fonctionnalités trop complexes.

. Dans le projet ComCoBB (Communication Coprocessor Building Block) de l'université UCLA [TAM 88], la conception et la réalisation VLSI des circuits de communication (en l'occurrence pour des communications asynchrones à haute vitesse entre processeurs) sont basées sur l'existence des modules spécifiques comme des FIFOs à haute performance, des barrières temporelles et une interface avec un bus microprocesseur. La démarche suivie par ce projet est comparable à la nôtre.

Note : Rappelons que l'état de l'art pour chaque classe de circuits de communication est présenté dans le chapitre correspondant.

#### **1.42. Domaines à problématique équivalente**

Dans le domaine du traitement du signal, en particulier pour traiter la parole, l'approche globale de processeur à usage général de traitement du signal (DSP) sera confirmée dans les années à venir [AOY 87] sous divers aspects. Les principaux efforts portent actuellement sur l'amélioration de la vitesse de traitement et sur l'aspect du génie logiciel inhérent à ce type d'architecture [CAN 88] [IME 88] : il s'agit d'optimiser l'exécution d'algorithmes de plus en plus sophistiqués. Cependant la problématique est la même que dans notre domaine de recherche : à côté des DSP, les circuits de traitement du signal à architecture dédiée à un type donné d'applications continuent à exister dans la mesure où le volume de production le justifie; sachant que les facilités de plus en plus grandes de conception des ASIC feront coexister encore longtemps les deux approches.

## Chapitre 2

### ARCHITECTURE

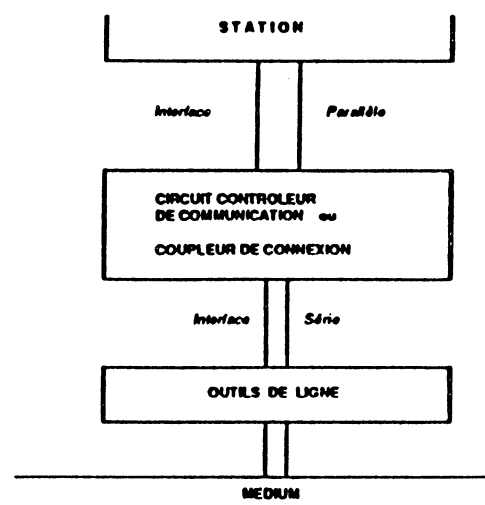
DES CIRCUITS DE COMMUNICATION POUR

LES COUCHES 2 ISO OU EQUIVALENTES

## 2.1. INTRODUCTION

Notre objectif est d'analyser, dans un premier temps, les différentes méthodes d'accès au medium de communication, qu'il soit parallèle ou série. Le terme méthode d'accès est à prendre dans le sens du terme Medium Access Control, MAC, du standard IEEE 802 (même s'il est spécifique aux réseaux locaux), c'est-à-dire celui d'une couche physique ou logique et de ses interfaces qui relie une station à son medium de communication, directement ou par l'intermédiaire des outils de ligne donnés.

Le circuit (ou la carte) qui implante le niveau MAC est appelé dans cet exposé un contrôleur de communication (respectivement coupleur de connexion).



Les analyses sont effectuées en terme de flot de données. En commençant par les besoins fonctionnels, le flot des informations à travers le système est analysé et les fonctionnalités principales déterminées. Les diagrammes des flots des informations sont développés et décomposés jusqu'à une profondeur suffisante pour que l'on puisse identifier les sous-systèmes ou processus principaux et les composants principaux de chaque sous-système [GOM 84]. Ainsi la synthèse de l'analyse peut-elle être effectuée à partir du moment où l'on est apte à identifier ces sous-systèmes et leurs composants.

Cette analyse s'accompagne de l'étude des architectures fonctionnelles des différents circuits intégrés existant implantant ces méthodes d'accès.

Ces analyse et étude débouchent sur une proposition de modèle d'architecture fonctionnelle générale pour chaque classe de méthodes d'accès, assortie de recommandations concernant les architectures matérielles des machines qui implantent ces méthodes d'accès.

Pour chaque classe ou sous-classe de méthodes d'accès, l'exemple de la conception d'une machine correspondante est décrit ou référencé.

## 2.2. VOIES PARALLELES

Les media parallèles sont dédiés aux communications qui s'établissent sur de courtes distances. Ce sont les cas des bus microprocesseurs, des bus fond-de-panier et des différents bus parallèles dédiés (à l'instrumentation, à la connexion des périphériques, etc...).

Leurs algorithmes de gestion d'accès multiples sont généralement à assignation dynamique par demande [TOB 86]. Le contrôle peut être centralisé ou décentralisé :

- . en contrôle centralisé, l'algorithme est exécuté par un arbitre unique physique ou logique. Dans le cas du bus VME analysé ci-après par exemple, il s'agit d'un arbitre unique physique;

- . en contrôle décentralisé, le rôle d'arbitre est joué par chaque entité d'un ensemble d'entités désignées *statiquement* à l'initialisation du système. C'est le cas des bus MULTIBUS 1 et SM90, analysés aussi dans le chapitre 2.221.

Le contrôle peut être totalement distribué (§ 2.222) : en mode distribué, l'algorithme est exécuté localement par *toutes* les entités; chaque demandeur sait individuellement s'il a gagné ou s'il a perdu. Ce n'est pas le cas des bus microprocesseur ou des bus multi-microprocesseurs, qui sont généralement basés sur la notion de maîtres-esclaves.

Précisons que les bus microprocesseur [GUS 84] ne seront pas analysés dans ce rapport. Cependant, les principales caractéristiques analysées ci-après peuvent être intégralement généralisées aux bus microprocesseurs.

### 2.21. Architecture fonctionnelle des coupleurs de connexion aux bus multi-microprocesseurs

#### 2.211. ANALYSE DES METHODES D'ACCES

L'analyse fonctionnelle des bus parallèles, en particulier des bus multi-microprocesseurs (bus VME ou IEEE P1014, MULTIBUS 1 ou IEEE 796, bus SM90), qui sont les cas les plus complexes, a permis de décomposer la gestion de ces bus en un ensemble de processus présentés ci-après.

- + Le processus *gestion de la demande d'accès* gère la demande d'accès au bus et la réception de la réponse à la demande.

- + Le processus *gestion d'accès* ou *arbitrage* gère le partage dynamique du bus. D'après l'analyse des différents arbitres de bus, les principales fonctions exécutées par les différents arbitres



sont les suivantes : arbitrage, retransmission du verdict et 'supervision'.

Les principales caractéristiques d'arbitrage suivantes ont été étudiées : arbitrage centralisé, arbitrage décentralisé, arbitrage visible et arbitrage caché. Il en a été de même pour les principaux algorithmes d'arbitrage : linéaire (ou en fonction de niveaux fixes des priorités [JOLY 83]), circulaire (ou en fonction de niveaux rotatifs de priorité) et mixte [OLI 82].

+ Le processus *gestion de l'adressage* permet de désigner une station avec laquelle la communication doit s'établir, ou permet à une station maître de s'identifier comme étant concernée par la communication. La désignation du correspondant utilise différents modes d'adressage et différents types d'adresses. L'établissement de la communication nécessite : la définition du type de communication adopté (lecture ou écriture), la gestion du transfert de donnée, la gestion de la synchronisation et la possibilité laissée à l'esclave de faire des comptes rendus d'erreur ou d'état.

+ Le processus *transfert de données* gère des signaux appelés bus de données pour véhiculer la valeur de la donnée. La souplesse dans la définition de l'adresse va de pair avec la possibilité de ne pas utiliser pour un transfert donné toute la largeur du bus de données, mais seulement un sous-ensemble. Cette possibilité est gérée soit de manière explicite (gérée par des signaux de contrôle spécifiques), soit de manière implicite (le maître possède la logique nécessaire pour s'adapter à telle largeur du bus de donnée).

Pour des méthodes d'adressage qui permettent l'utilisation d'adresses dont la valeur ne correspond pas à un multiple de la taille du bus de données, si le mécanisme d'adressage du bus n'offre pas la souplesse voulue, la gestion de l'écriture d'une donnée quelconque doit éventuellement se décomposer en plusieurs cycles d'opérations élémentaires : lecture, modification, écriture.

Mentionnons les possibilités laissées dans la conception de certains bus du multiplexage adresse et donnée, du contrôle d'erreur et de la correction d'erreur. La gestion de la longueur du transfert est une autre fonctionnalité complexe que l'on doit traiter, soit de manière explicite, soit de manière cachée en gérant directement les signaux de contrôle du bus : cette fonctionnalité est à la frontière des problèmes communs aux bus parallèles et aux réseaux locaux.

+ Le processus *gestion de la synchronisation* permet de contrôler la transmission d'une information, donnée ou adresse. Il est nécessaire car, dans la mesure où les informations voyagent sur des lignes parallèles, il est électriquement impossible d'assurer que la vitesse de transmission sera la même sur chaque ligne; ce, au vu des caractéristiques électriques des émetteurs, des récepteurs et indépendamment du type de signal utilisé : trois-états, collecteur-ouvert pour le TTL, drain-ouvert pour les MOS, etc ...

Les aspects de transfert synchrone (lié à une fréquence d'horloge maximale donnée) et de transfert asynchrone (lié à la notion d'événement) sont souvent mélangés dans la conception des bus. Notons la plus grande complexité de la gestion de synchronisation des transferts asynchrones.

D'autre part, et en particulier en environnement multi-microprocesseurs, ce processus gère les interruptions et les niveaux d'interruption entre les différents maîtres et les esclaves.

+ Le processus *service* gère les fonctions de service telles que alarme, initialisation, détection d'erreurs, 'supervision' (nous considérons la fonction de supervision comme une fonction de détection d'anomalie, donc faisant partie du processus *service*).

Note : Cette analyse nous aura été fort utile lors des choix qui sont intervenus dans la spécification de l'architecture interne du composant MC3 et de ses interfaces avec l'univers extérieur (§ 3.22).

On observe que les cartes coupleur et les circuits contrôleur de communication qui gèrent les méthodes d'accès aux bus multi-microprocesseurs [BAR 84] sont bâtis à partir d'un ensemble de circuits intégrés (ou de modules faisant partie des circuits intégrés) qui implantent les processus analysés ci-dessus.

## 2.212. ARCHITECTURE FONCTIONNELLE DES CARTES COUPLEURS DE CONNEXION OU DES CIRCUITS CONTROLEURS DE COMMUNICATION

Le modèle d'architecture fonctionnelle que nous proposons pour cette classe de méthodes d'accès est un réseau d'unités asynchrones avec les caractéristiques suivantes :

- . Chaque unité exécute un processus déterminé.
- . Le nombre d'unités est statiquement défini à la conception.
- . Les unités ne partagent pas de mémoire commune.
- . Il n'y a pas d'effet de "pipe-line" au niveau du flot de données.
- . Le parallélisme entre unités est effectif.
- . La communication entre les unités est faiblement couplée et ne concerne que les informations de commande.
- . Si le contrôle central du fonctionnement du modèle est nécessaire, il est généralement de type maître-esclave : un processus maître dirige l'ensemble des processus esclaves.

Ce modèle d'architecture peut être utilisé, par raffinement successif, pour la définition de l'architecture fonctionnelle de l'unité gestion d'accès (ou arbitre).



## 2.22. Travaux réalisés

### 2.221. ETUDE DE L'ARCHITECTURE ET CONCEPTION D'UN CIRCUIT ARBITRE DE BUS MULTI-PROTOCOLES : ABCM

#### Etude système

Nous nous sommes intéressés à la compatibilité de l'arbitre de bus intégré ABC 90 avec les autres types de bus : MULTIBUS 1 et VME, dans la mesure où les fonctionnalités de l'ABC 90 sont plus riches que celles offertes par les circuits intégrés réalisant l'arbitrage sur les autres bus (priorités fixes, contrôle centralisé) [BAR 84]. Mentionnons, parmi d'autres avantages, les possibilités de gérer des priorités cycliques, les mécanismes de contrôle d'erreur et le contrôle décentralisé.

La première étape de ce projet consistait à étudier l'utilisation de l'ABC 90 comme organe d'allocation du bus dans différentes configurations d'architecture MULTIBUS 1 et VME, et ce par adjonction d'éléments discrets, sachant que les compatibilités électrique et mécanique ont été vérifiées et ont nécessité dans certains cas la re-définition de quelques connecteurs et l'adjonction de quelques signaux supplémentaires. De nouvelles architectures *discrètes* ont été proposées pour les bus MULTIBUS 1 et VME :

- + architecture centralisée avec des niveaux rotatifs de priorité et supervision des arbitrages rendus,
- + architecture décentralisée avec plusieurs arbitres en priorité fixe ou tournante et supervision de l'arbitrage.

Un exemple caractéristique de ces nouvelles architectures compatibles proposées est l'architecture présentée ci-dessous qui permet de faire un arbitrage décentralisé sur un bus VME, bus primitivement défini à contrôle centralisé.

Cette architecture dispose :

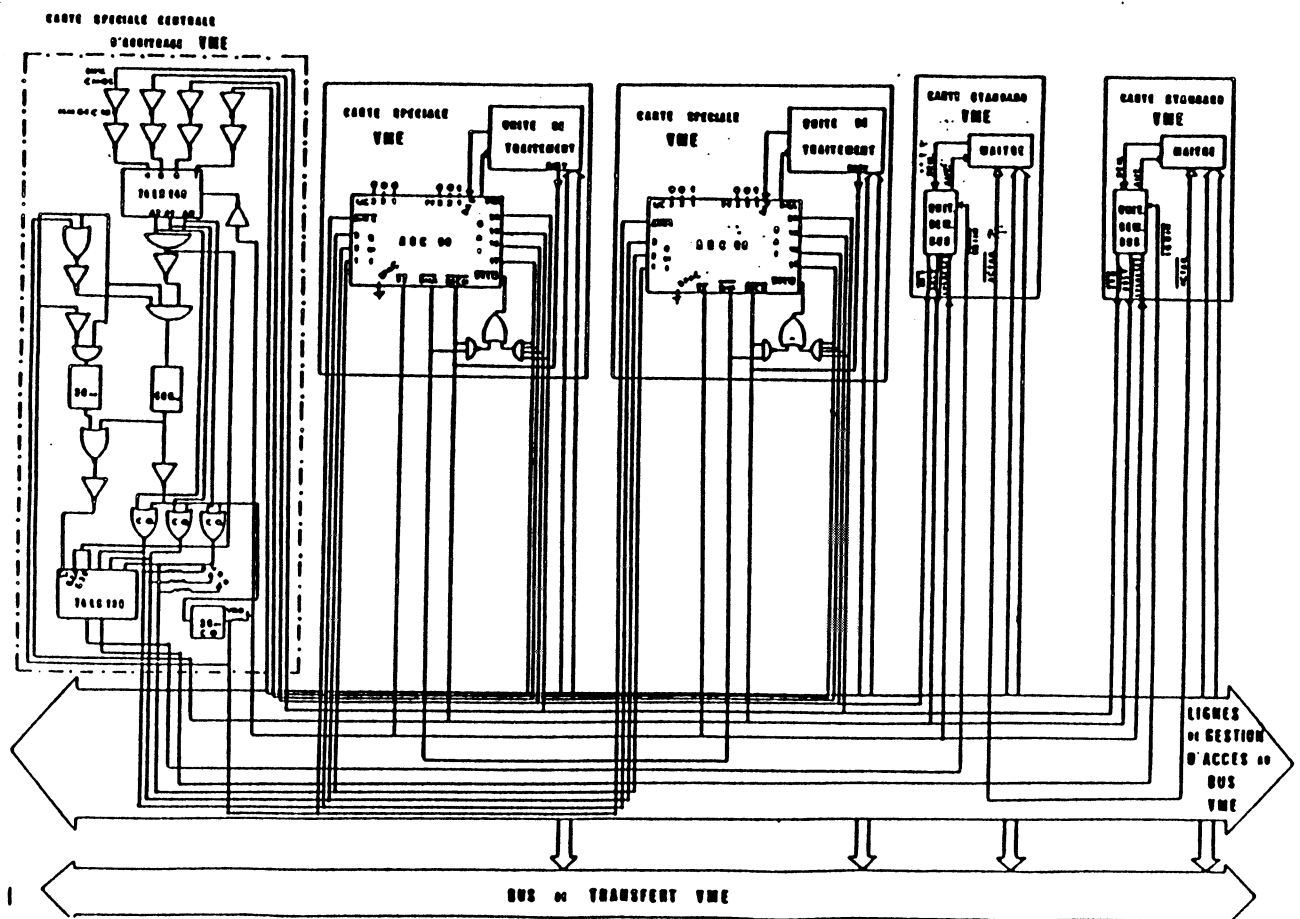
- . d'un ABC 90 sur chaque carte maître à la place de l'unité de demande du bus VME et utilisant toutes les potentialités : affichage, arbitrage, retransmission du verdict et supervision; on obtient dès lors plusieurs ABC 90 travaillant en parallèle, de manière décentralisée;
- . des cartes standard VME, ne contenant pas l'ABC 90 et
- . de la logique centralisée d'arbitrage réalisant un algorithme fixe de priorités dès qu'une unité de traitement standard VME prend le bus.

Les avantages d'une telle architecture résident dans :

- . la liberté totale de configuration de chaque unité de traitement,
- . la modularité et la disponibilité, les arbitrages parallèles étant rendus par des organes implantés sur des unités de traitement distinctes et
- . l'incorporation de mécanismes de contrôle et de récupération des erreurs grâce à la supervision.

Les inconvénients proviennent d'une part du coût du matériel additionnel et d'autre part d'un relatif ralentissement du bus (dans le meilleur cas, quand l'arbitrage est rendu par un ABC 90, il est de l'ordre de 200 ns).

La validation de ces différentes architectures a été effectuée par simulation logico-fonctionnelle avec un langage de type HDL : FIDEL [HAZ 84].

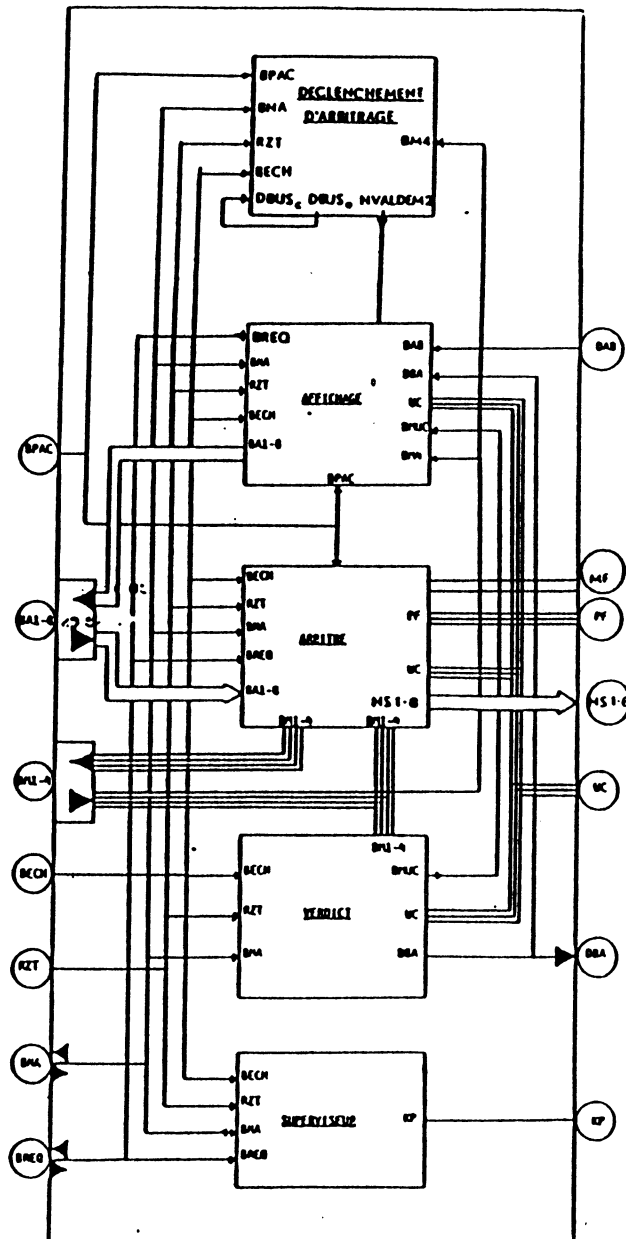


Architecture VME : arbitrage décentralisé [BAR 84].

## Architecture fonctionnelle de l'ABCM

La conception du circuit intégré arbitre de bus multi-protocoles, qui a constitué la deuxième étape du projet, a été effectuée en partant de la définition de l'ABC 90 et y intégrant les *modules conçus en éléments discrets* dans la première étape du projet.

Cette approche du problème implique que le choix de l'architecture fonctionnelle respecte celui du composant ABC90, comme le montre la figure suivante de l'architecture fonctionnelle du nouveau composant ABCM.



Architecture du circuit ABCM [BAR 84].

Les modules obtenus par modification ou nouvellement introduits (en particulier le décodage du verdict et le déclenchement de l'arbitrage), ainsi que l'architecture complète du nouveau circuit ont été simulés logiquement au niveau des portes de base avec le simulateur logique EPILOG.

## Faisabilité technologique et Conception du circuit

La technologie employée, le NMOS L3, était une technologie éprouvée et imposée de fait par le circuit ABC 90.

Le module 'décodage du verdict', dont l'algorithme a été modifié pour divulguer systématiquement le verdict, a été implanté sous forme de partie opérative. Il a été dessiné directement, vu sa simplicité, sous l'éditeur STICK du système de CAO du CNET : CASSIOPEE [BEY 82]. Sa complexité est en effet réduite : une bascule, un sélecteur, un décodeur 3X8, un registre maître-esclave de 8 bits et un générateur d'horloge non recouvrante. Par contre, la synthèse du nouveau module de 'déclenchement de l'arbitrage' a été effectuée avec la méthode de synthèse développée pour la conception du circuit ABC 90 [OLI 83].

Le nouveau circuit ABCM était obtenu à partir de l'ancien au prix d'un nouveau placement des différents blocs et d'un nouveau routage. Le nouveau plan de masse est supérieur de l'ordre de 20% en surface par rapport à l'ancien.

## Conclusions

La conception du circuit ABCM [BAR 84] a permis d'une part d'offrir des fonctionnalités système supplémentaires au niveau de l'arbitrage des bus VME et MULTIBUS 1, au prix d'une baisse de performance minimale comme témoigne le tableau de synthèse suivant :

<b>CARACTERISTIQUES</b>		<b>MODES DE FONCTIONNEMENT</b>			
<b>D'UTILISATION DE L'ABCM</b>		<b>ARBITRAGE DECENTRALISE</b>			
	<b>SM90</b>	<b>VME</b>		<b>MULTIBUS</b>	
		(avec l'ABCM en "Release When Done")		(avec l'ABCM en "Release on Request")	
Temps d'arbitrage :	240 ns	250 ns		250 ns	
Type d'arbitrage :	Visible ou Caché	Visible		Caché	
Supervision :	Optionnelle	Optionnelle		Nécessaire	
Algorithme d'arbitrage :	Fixe, Rotatif ou Mixte	Fixe, Rotatif ou Mixte		Fixe, Rotatif ou Mixte	

Elle nous a permis d'autre part de nous conforter dans l'idée que si l'implantation de la machine cible (le circuit arbitre de bus multi-protocoles) respecte les principaux concepts de l'architecture fonctionnelle, tels qu'ils ont pu être analysés dans le paragraphe 2.21, le rajout ou la suppression de fonctionnalités peut être faite de manière modulaire, efficace dans le temps et en ce sens plus sûre qu'une nouvelle conception effectuée en repartant d'un nouveau cahier des charges.

L'ensemble de ces travaux a abouti à la synthèse du circuit arbitre de bus multi-protocoles ABCM en NMOS-L3 [BAR 84]; cette synthèse a été effectuée dans la division CCI du CNET CNS.

Note : Les études des bus fond-de-panier (en particulier VME), des architectures multimicroprocesseurs et l'étude de l'architecture des circuits arbitres de bus ont été mises à profit lors de la conception des bus système internes de MC3 et de celle des environnements système du composant MC3 (§ 3).



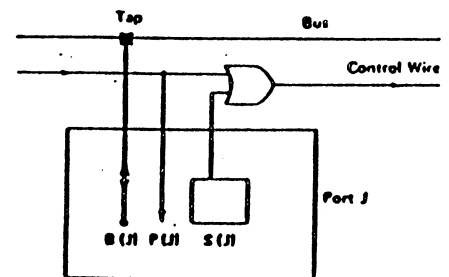
## 2.222. ETUDE DE L'ARCHITECTURE ET CONCEPTION D'UNE CARTE COUPLEUR POUR LA CONNEXION DE MICRO-ORDINATEURS HETEROGENES A UN BUS PARALLELE A CONTROLE DISTRIBUE

### Introduction

Les algorithmes de gestion d'accès multiples à une voie de communication par assignation dynamique peuvent être divisés en deux sous-ensembles [TOB 86] : algorithmes à base de contention ou de compétition [COR 81], qui ne sont en fait utilisés que dans le cas des bus série et des réseaux locaux (§ 2.31) et algorithmes à assignation par demande, qui sont largement utilisés dans le cas des bus parallèles (§ 2.221) ainsi que dans le cas des bus série.

Un exemple caractéristique de bus parallèle à assignation par demande avec un algorithme *distribué* de gestion d'accès est donné par l'algorithme de Eswaran, Hamcher et Shedler défini dans [ESW 81], appelé ci-après E.H.S., et dont des implantations ont été réalisées par [FRA 83], [DAN 83].

L'algorithme suppose l'existence d'une ligne de contrôle qui reprend l'idée classique d'une ligne de OU câblé. Soit  $J$  la station.  $P(J)$  reçoit le signal des OU câblés des stations de gauche.  $S(J)$  permet à la station de pouvoir demander le bus. L'algorithme de E.H.S. tient compte du temps  $T$  total de propagation sur le bus et du temps  $R(J)$  de propagation du signal de la station  $J$  jusqu'à la station la plus à droite du bus.



L'algorithme de E.H.S. est le suivant :

- . Attendre jusqu'à ce que l'on trouve le bus libre pendant un intervalle de temps  $2 * T$
- . Positionner  $S(J)$  à 1
- . Attendre un délai  $R(J) + T$
- . Attendre jusqu'à ce que le bus soit libre et  $P(J) = 0$
- . Commencer à transmettre en remettant simultanément  $S(J)$  à 0

Les avantages de cet algorithme de gestion d'accès à un bus sont les suivants :

- . Le contrôle est totalement distribué.
- . Le temps d'accès est borné supérieurement, une preuve théorique a été fournie dans [HAM 81].
- . Il y a un partage équitable de la prise de contrôle du bus entre les stations.
- . Il n'y a pas d'hypothèses faites sur la gestion de l'adressage, du transfert des données et

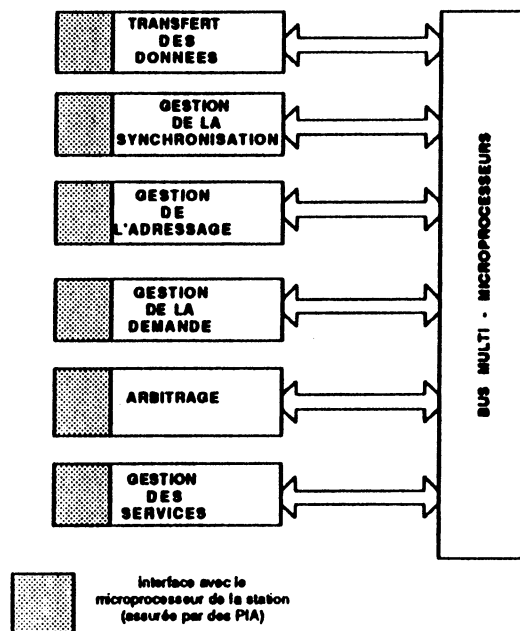
de la gestion de la synchronisation. Hypothèses dont nous avons le libre choix de définition en vue de l'implantation.

### Etude système

Nous avons entièrement spécifié et réalisé, en utilisant l'algorithme de E.H.S., un petit réseau local, appelé SLAN, pour connecter des micro-ordinateurs hétérogènes [DAN 83]. Le réseau local utilise un medium parallèle, supposé fiable et travaillant dans un environnement non hostile. Les services de communication offerts à un utilisateur sont ceux de la diffusion, de la sous-diffusion et du point-à-point. Les domaines potentiels d'utilisation sont les suivants : support technologique de cours sur les réseaux d'ordinateurs (les principaux concepts de la communication se trouvent réunis), prototype pour la connexion des équipements et des micro-ordinateurs personnels, essentiellement hétérogènes.

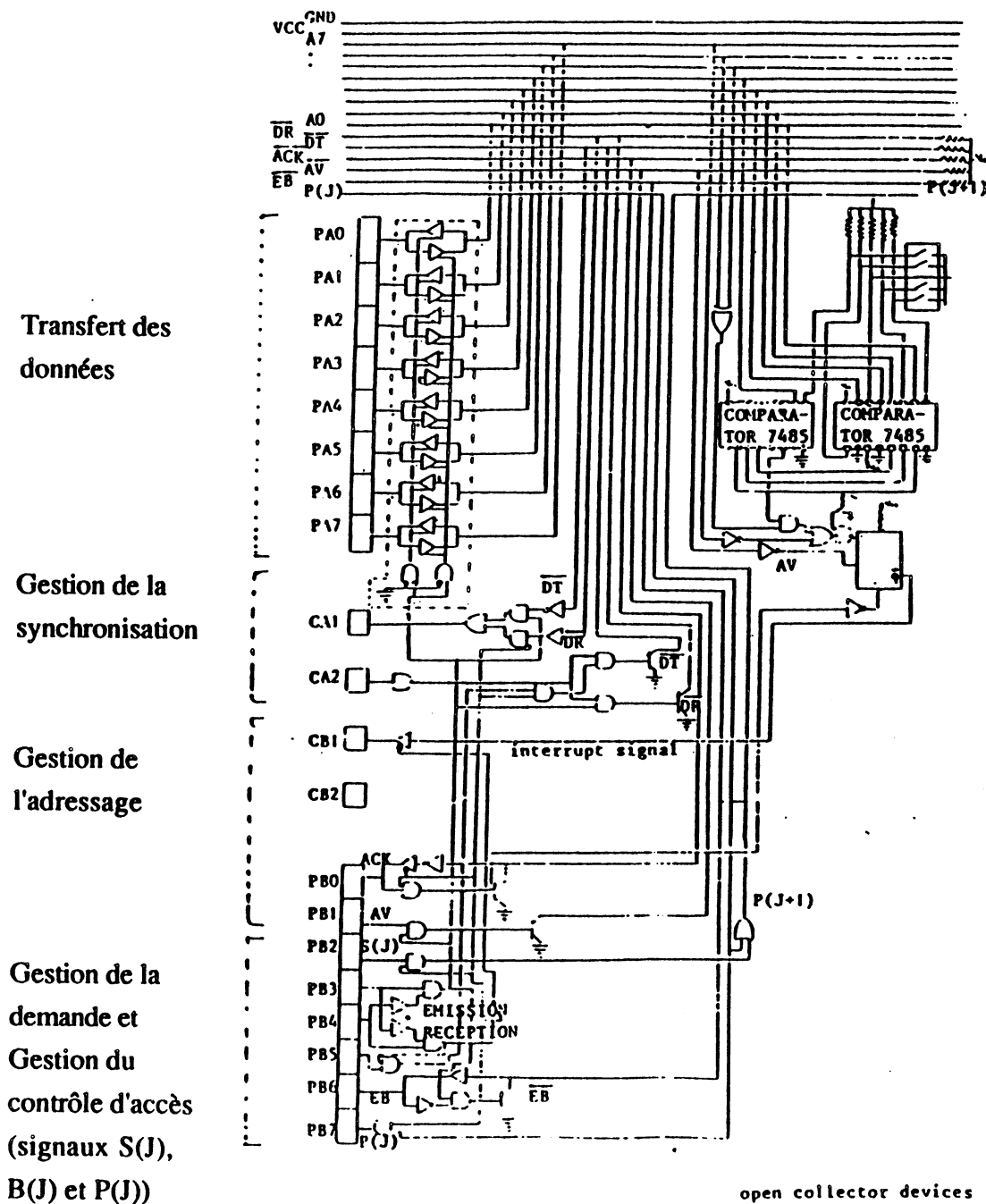
L'architecture fonctionnelle de la carte coupleur respecte le modèle présenté dans le paragraphe 2.21. Les principaux choix effectués lors de conception de la carte coupleur sont décrits dans [DAN 83].

L'interface de la carte coupleur avec la station (transformation des signaux électriques en information binaire et vice versa, existence d'un étage tampon de bascules, gestion du vecteur d'interruptions) est assurée par des composants standard du type PIA [DAV 80] ou VIA.



L'utilisation modulaire du composant PIA 6821 ou VIA 6522 permet d'une part de rendre la carte indépendante des microprocesseurs choisis. Elle permet d'autre part de rendre la carte commandable à partir de l'exécution d'instructions du microprocesseur. Ainsi toutes les fonctions de la carte sont-elles commandées ou suivies au niveau des programmes fonctionnant sur le microprocesseur hôte.

Le schéma logique suivant du coupleur de connexion au réseau local SLAN donne un aperçu de son absence de complexité intrinsèque.



Coupleur de connexion au réseau SLAN en technologie TTL-LS

## Conclusion

Trois prototypes de ce coupleur non-intelligent de communication ont permis l'implantation d'un petit réseau local entre deux micro-ordinateurs à base de microprocesseur 6502 et un micro-ordinateur EXORCISER à base de microprocesseur 6800. Ils ont permis d'autre part la validation de l'algorithme de E.H.S. Le débit de transmission était de l'ordre de 130.000 bits/seconde, débit conséquent par rapport à ceux des bus industriels de bas de gamme.

Cette implantation a surtout permis d'évaluer la complexité d'un tel coupleur de communication. La complexité globale du coupleur (hors PIA) est évaluée à 825 transistors équivalents en technologie TTL-LS. Il ne possède ni CPU (c'était le cas des coupleurs des bus de contrôle I<sup>2</sup>C et D<sup>2</sup>B de Philips [GOL 82], utilisés comme bus pour la connexion d'équipements divers, dont la deuxième génération de circuits contrôleur sont actuellement encore conçus et fabriqués [MHS 87]), ni de mémoire RAM ou de partie opérative complexe, ni de ROM de micro-instructions (comme c'est le cas du coupleur de bus parallèle du réseau NESTAR Systems Cluster/One pour la connexion des micro-ordinateurs APPLE II, aussi à base de microprocesseur 6502).

Signalons que l'architecture implantée du coupleur de connexion respecte le découpage défini par l'architecture fonctionnelle. Nous attribuons en partie la rapidité et la sûreté avec laquelle la carte a été conçue et testée à ce facteur.

## Bus parallèle et Réseaux Locaux

Le bus parallèle tel que nous l'avons implanté se rapproche effectivement des caractéristiques fonctionnelles des réseaux locaux [CLA 78] [SHO 80]:

- . Il autorise la connexion et la déconnexion dynamique de station, sans perturbation pour le bus en cours de service.

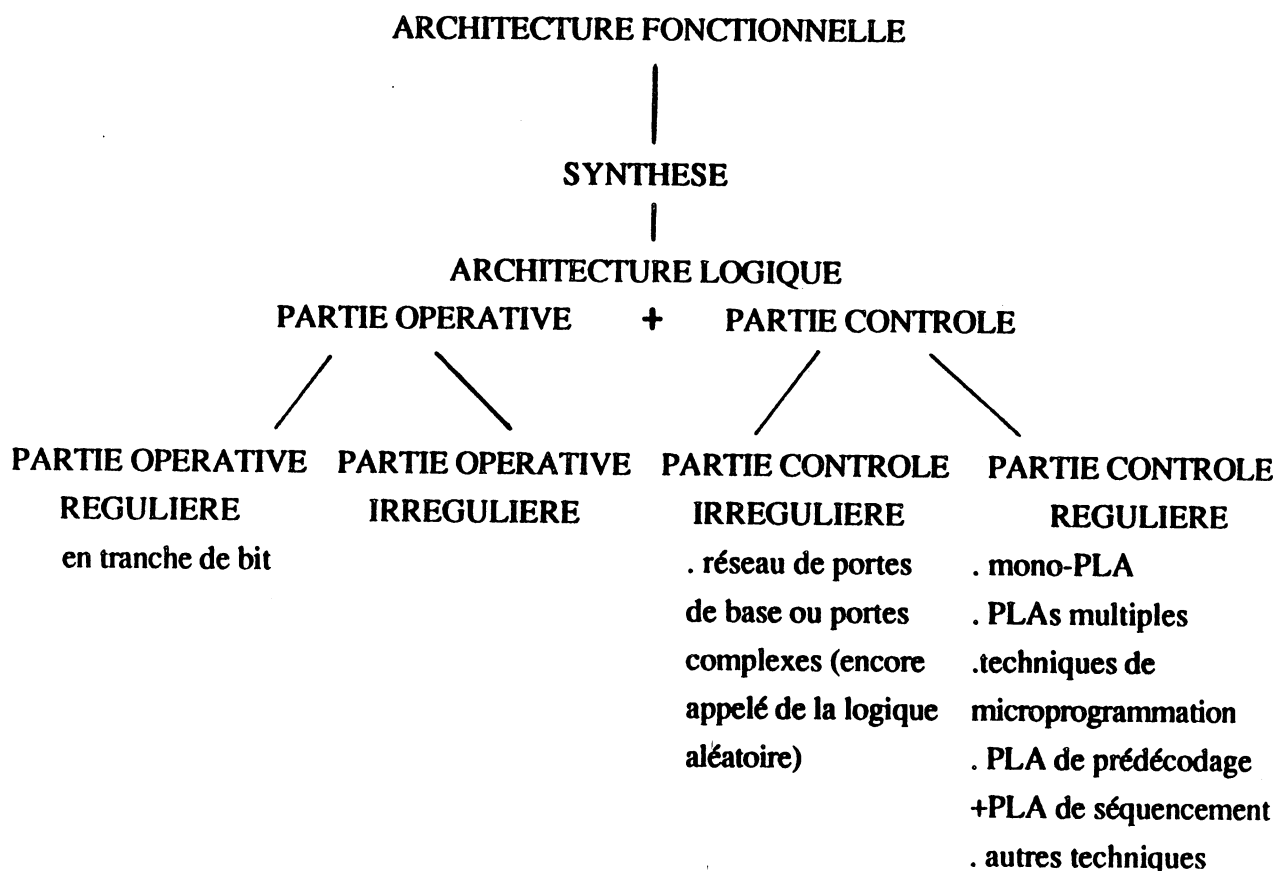
- . Il autorise les transferts de messages de longueur variable.

- . En utilisant l'interface PIA pour communiquer avec le microprocesseur hôte, il constitue un proposition d'interface standard utilisable par une grande variété de microprocesseurs et micro-ordinateurs, des équipements de contrôle et d'autres composants électroniques.

## 2.23. Recommandations pour les choix d'architecture matérielle

Le passage d'une architecture fonctionnelle en architecture matérielle, c'est-à-dire en architecture logique puis en architecture implantée, est appelé par les architectes et concepteurs de circuit une *synthèse*. Les différentes étapes de la synthèse (passage en architecture logique puis passage en architecture implantée) sont effectuées soit manuellement, soit plus ou moins automatiquement selon les outils de CAO dont on dispose.

Nous avons choisi, à l'instar d'autres architectes, de décrire nos architectures sous la forme d'une *partie opérative* et d'une *partie contrôle*. Les méthodes de synthèse existantes, qui diffèrent en fonction de l'architecture cible, peuvent être résumées par la figure suivante :



Note : Le terme de PLA signifie tableau logique programmable. Il sert à calculer q fonctions booléennes en fonction de p variables d'entrée; il est constitué par deux matrices de transistors, une matrice 'produit' et une matrice 'somme'.

## CHOIX DE L'ARCHITECTURE CIBLE

Le choix de l'architecture cible dépend de plusieurs contraintes [REY 84] : contraintes temporelles, contraintes électriques, contraintes d'optimisation dans l'utilisation de ressources, contraintes de place, contraintes dues à la présence ou l'absence d'outils de synthèse automatique. *Un des progrès* dans la conception des outils de CAO pour rendre la situation de choix plus confortable pour un utilisateur système sera de pouvoir disposer d'outils de génération automatique en fonction de différentes architectures cible possibles, de pouvoir effectuer des simulations de génération d'architecture et de comparer les résultats de ces simulations pour affiner le choix.

## RECOMMANDATIONS

L'évaluation des principales fonctions des méthodes d'accès des bus multi-microprocesseurs connus et des bus parallèles a montré le degré moyen de complexité des circuits contrôleur de communication ou des coupleurs de connexion qui les implantent.

Notons cependant que dans le cas des bus parallèles et microprocesseurs, les *contraintes temporelles doivent être considérées comme contrainte prioritaire*. Le respect fin des performances est vu comme un préalable important à la synthèse de ces fonctions et influe de ce fait sur les choix d'architecture cible, au contraire d'autres cas de synthèse où la performance n'est que le résultat tiré d'une architecture cible générée ou choisie en fonction d'autres contraintes plus prioritaires.

L'approche que nous préconisons pour implanter les coupleurs de connexions (coupleur du réseau SLAN) et les circuits contrôleur de communication (ABC-M) qui gèrent les méthodes d'accès aux bus parallèles consiste à faire la synthèse de chaque processus (ou de chaque unité asynchrone), indépendamment les uns des autres.

Ainsi, pour les parties contrôle des unités asynchrones dont les contraintes temporelles sont élevées, cette synthèse peut-elle se faire avec un réseau de portes de base ou portes complexes (dit synthèse en logique aléatoire); il s'agit généralement du type d'architecture cible le plus performant du point de vue de l'optimisation du temps de réponse.

Il existe des générateurs automatiques de partie contrôle en logique aléatoire (structure irrégulière) qui proposent la prise en compte des contraintes de génération telles que des contraintes temporelles ou des contraintes électriques [APT 88].

Mentionnons la proposition intéressante de synthèse par un réseau d'automates asynchrones développée par [OLI 83][OLI 87] pour le circuit arbitre de bus ABC 90 de la SM90. Cette synthèse vers une structure régulière a pour origine paradoxalement la même motivation technique : résoudre des contraintes de temps de réponse qui étaient à la limite de la technologie employée.

## 2.3. VOIES SERIES

### 2.31. Introduction

Les communications 'série' entre ordinateurs sur de courtes distances (jusqu'à quelques kilomètres) ont connu des développements extrêmement diversifiés depuis les années 1975 [CLA 78][SHO 80][POW 81][LeL 84][TOB 86]. L'émergence de différentes classes de réseaux locaux a été analysée dans [DAN 82].

#### a) Les réseaux locaux industriels.

Ils sont destinés à supporter des applications de régulation, de contrôle-commande et des applications dont les contraintes temps-réel d'utilisation sont considérées comme critiques [LeL 83]. La décomposition en quatre niveaux est généralement la plus acceptée : (A) à l'intérieur d'un robot ou d'une machine, (B) entre les différentes machines - robots - capteurs - actionneurs, (C) pour les systèmes de supervision et (D) entre les divers systèmes [HER 87][GAL 84]. A chaque niveau, correspond une classe de réseaux qui possède ses propres caractéristiques : classe de débit, type de méthode d'accès. Le medium est généralement un *bus série* à accès multiple.

Les méthodes d'accès caractéristiques des réseaux de la classe B et de la classe C sont soit les méthodes à assignation par demande (bus FIP, BITBUS, bus 1553 B), soit les méthodes à arbitrage au vol sur un bit (principe du réseau LISA [MAR 80], réseau CAN [BOS 87]). Une analyse des contraintes fonctionnelles de ces classes de réseaux locaux industriels a été effectuée dans [DAN 86.1], ainsi qu'une analyse des fonctionnalités et des circuits contrôleur de communication des principaux bus et réseaux locaux industriels.

Les réseaux locaux industriels de la classe D sont destinés à interconnecter des ordinateurs. Deux exemples de méthodes d'accès caractéristiques des réseaux de cette classe sont présentés ci-dessous :

. La méthode basée sur le standard IEEE 802.4 : c'est le cas du projet MAP, initié et soutenu par un ensemble d'utilisateurs dont General Motors [KAM 86], [ESP 86] qui se présente comme un standard pour les réseaux locaux industriels. Le réseau local MAP, en ce qui concerne les couches basses (jusqu'au niveau MAC), est un bus série large bande dont la méthode d'accès est basée sur le principe de l'assignation sur demande (l'algorithme du jeton sur bus).

. La méthode d'accès basé sur la contention mais à résolution de conflits déterministe (CSMA/DCR, voir paragraphe § 2.332) : ce sont les cas, par exemple, des réseaux locaux FACTOR [APT 87.1] et RECITAL [ESD 88].

Note : Le projet de réseau CICS 81 [MIC 82], [DAN 82], qui a motivé notre collaboration avec le département Architecture des Micro-Systèmes du CNET CNS, faisait aussi partie de cette classe D de réseaux, de même que le réseau local EXPRESSNET [FRA 81] qui propose le même algorithme de gestion d'accès au réseau que CICS 81.

**b) Les réseaux locaux bureautiques ou d'entreprise.**

Cette classe de réseaux locaux vise à offrir une variété de services aux utilisateurs dont le transport des informations, de la voix, sinon du transfert d'images soit fixes soit animées. Citons deux types de réseaux de référence :

- . les réseaux basés sur une méthode d'accès à base de contention à résolution non déterministe de conflits (appelée encore algorithme d'accès par assignation statistique), dont le plus connu est CSMA/CD et

- . les réseaux basés sur une méthode d'accès par assignation sur demande (par exemple, l'anneau à jeton d'IBM, l'anneau à fibre optique FDDI [DYK 88]).

Le réseau LCT 6500, basé sur l'expérience CARTHAGE développée au CCETT de Rennes [REN 82], est un excellent exemple de cette classe. Ce réseau, dont la topologie est un anneau, utilise une méthode d'accès par assignation mixte (temporelle et sur demande).

## **2.32. Architecture fonctionnelle des circuits contrôleur de communication des réseaux locaux**

### **2.321. LE STANDARD IEEE 802**

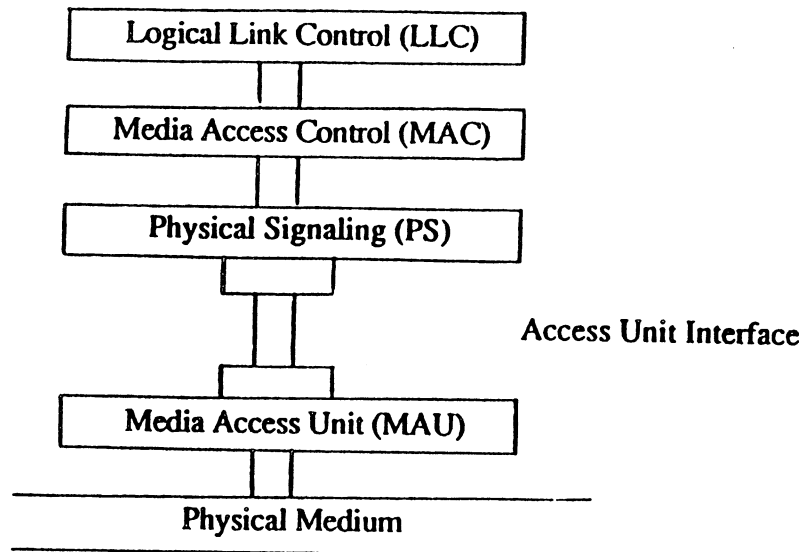
Le standard IEEE 802 propose le découpage suivant des couches 1 et 2 de la norme OSI (voir figure ci-dessous). Rappelons que la couche qui gère les méthodes d'accès au médium est la couche MAC.

Les frontières proposées par ce découpage sont très proches des choix généralement adoptés par les concepteurs de cartes coupleur de connexion pour un réseau local donné. Comme nous l'avons mentionné dans l'introduction, nous utilisons dans la suite de cet exposé ce découpage et sa terminologie pour décrire les circuits intégrés de communication.



### 2.322. ARCHITECTURE FONCTIONNELLE

Une analyse complète des architectures des circuits de communication de la couche MAC de l'IEEE 802.3 (INTEL 82586, LANCE 7990, etc ...) a été effectuée dans le cadre de la thèse [VOT 86]. Nous avons réalisé une analyse complémentaire en tenant compte des nouveaux circuits MAC 802.4 (MOTOROLA 68824 [MOT 86]) et MAC 802.5 (TI LAN Adapter [MOK 84]).



Les niveaux du standard IEEE 802 sur les Réseaux Locaux

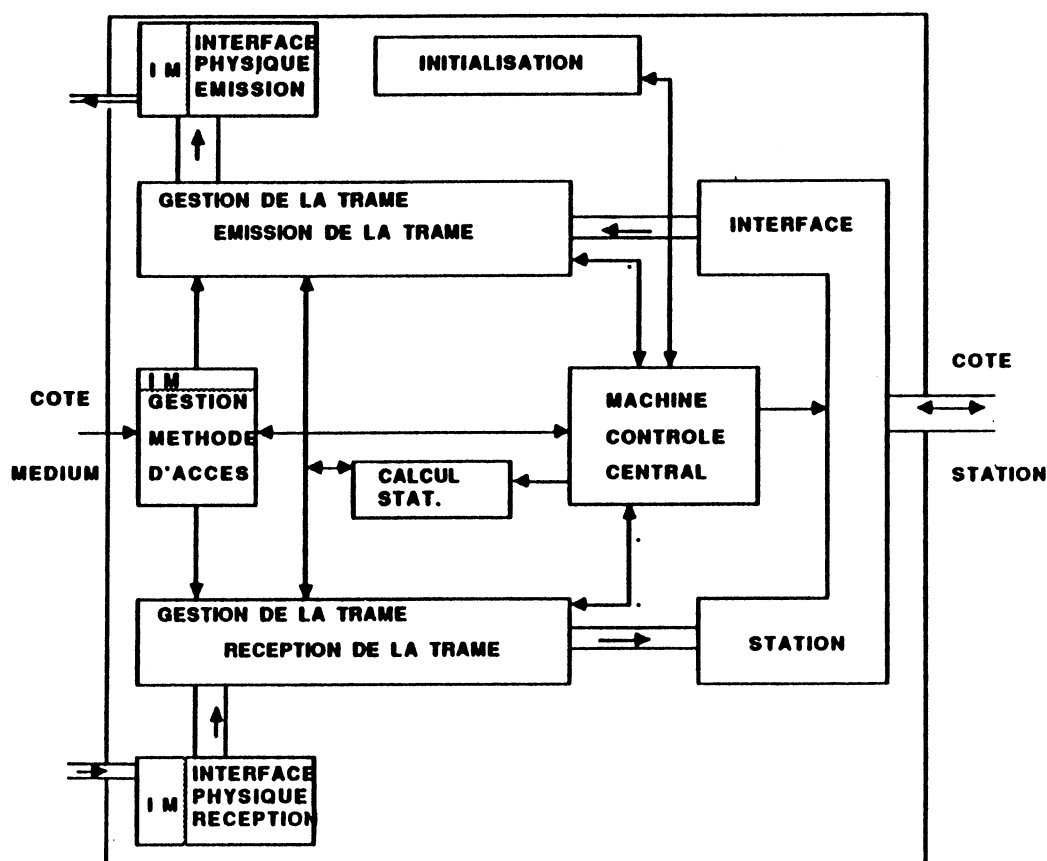
La synthèse de cette analyse, ainsi que notre propre expérience de conception du circuit FIP-VLSI et de l'étude CSMA-DCR (§ 2.33 Travaux réalisés) nous conduisent à proposer le *modèle d'architecture fonctionnelle* suivant : le modèle est constitué par un ensemble des processus qui est schématisé et décrit ci-dessous. Cet ensemble implante la gestion des méthodes d'accès aux bus série. Les caractéristiques du modèle sont présentées dans le paragraphe § 2.323.

+ Le processus IS, *Interface avec la Station*, gère la communication avec la station qui peut être une station basée sur un microprocesseur ou une station câblée. L'IS gère le médium de communication (généralement le bus mémoire du microprocesseur ou un bus parallèle dédié) qui permet à la station de communiquer avec le circuit contrôleur.

Le rôle d'un tel processus qui gère la méthode d'accès à un support de communication parallèle a déjà été analysé dans le paragraphe 2.21. Par conséquent, l'architecture fonctionnelle du processus IS peut être décomposée en un ensemble de processus ou d'unités asynchrones, tels qu'ils ont été présentés dans ce paragraphe : gestion de la demande d'accès, gestion d'accès, gestion de l'adressage, transfert de données et gestion de la synchronisation (dont la gestion des interruptions) et gestion des services [PIL 88].

Le processus IS, et par incidence le circuit contrôleur de communication, peut être vu par la station soit comme un esclave soit comme un périphérique, i.e. il existe dans l'IS des registres adressables en lecture et en écriture (pour passage de commandes et affichage de résultats ou d'états); c'est le cas des circuits MC 68824, LANCE 7990, MB 8795A, DP 8390. Inversement, la communication entre la machine IS et la station peut être faiblement couplée; ce principe est adopté par INTEL pour ses circuits de communication, par exemple INTEL 82586, 82588.

Notes : Le processus IS gère en fait une autre interface; il s'agit de l'interface interne entre lui-même et les autres processus du circuit. Comme le support de communication de cette interface est soit un bus en point-à-point, soit un bus en multipoint, nous ne jugeons pas utile de développer l'architecture fonctionnelle de cette interface.



#### Modèle de l'architecture fonctionnelle d'un contrôleur de communication de niveau MAC

Pour s'adapter à plusieurs familles de microprocesseurs, les circuits intégrés contrôleur de communication proposent soit de prendre en compte la logique additionnelle dans le processus IS, soit des adaptations externes sous forme de séquenceurs câblés [AMD 85] [VOT 86] ou sous forme de composants existant sur le marché; ces deux possibilités ne sont pas exclusives.

+ Le processus IM, *Interface avec le medium*, gère la communication avec les outils de ligne. Le rôle d'un tel processus qui gère la méthode d'accès à un support de communication série peut être comparé avec les conclusions de l'analyse du paragraphe 2.21. Les processus suivants sont à gérer en fait : gestion de la demande d'accès, gestion d'accès, transfert de la trame, gestion de la synchronisation (la communication avec les outils de ligne sont, dans la plupart des cas, synchrones; l'horloge étant fournie par les outils de ligne) et gestion des services. Les deux premiers processus dépendent étroitement de la méthode d'accès choisie.

Dans notre analyse qui est circonscrite au niveau MAC, nous supposons que les fonctionnalités de transformation analogique-digital et vice-versa, ainsi que des fonctionnalités associées aux premières, sont réalisées par les outils de ligne (telles que encodage, décodage, génération de préambule, suppression de préambule, ...).

Note : La gestion de la redondance du medium [APT 87.1], la gestion du multiplexage (plusieurs contrôleurs MAC se partageant la même connexion sur le medium) et la gestion des fonctions spécifiques du réseau qu'il est nécessaire de traiter à ce niveau, sont d'autres facteurs qui influent sur la définition de l'IM.

+ Le processus *Gestion de la Trame* est décomposable en deux autres processus : Réception d'une Trame et Emission d'une Trame.

Le processus Réception est défini par le sous-ensemble de processus suivants :

. Communication avec l'Interface Medium

- \* Barrière temporelle
- \* Gestion du bus de communication avec l'Interface Medium

. Traitement en temps-réel des éléments du Protocole dont la sémantique autorise la prise en compte et le traitement par le circuit de communication, par exemple :

- \* Reconnaissance de différents types d'adresse et des adresses
- \* Reconnaissance de suites constantes de valeurs binaires (par exemple, fanions spécifiques)
- \* Traitement spécifique tel que Horloge Réseau par exemple

. Transfert des informations (adresse, donnée, commande) vers la station

- \* Gestion du bus de communication avec l'Interface Station

Le transfert des informations vers la mémoire via un bus parallèle que le circuit de communication partage avec d'autres composants nécessite l'existence des :

- \* Processus Accès à la Mémoire et
- \* Processus Tampon.

. Traitement des commandes de la station pour le processus Réception de Trame

- \* Processus Accès aux informations des commandes
- \* Gestion du bus de communication avec l'Interface Station, si ce bus existe
- \* Traitement des commandes (par exemple : Modification dynamique des adresses locales au processus Réception, Chargement de paramètres)

. Gestion des erreurs

- \* Contrôle d'erreur (par exemple : CRC, Bit de parité, Suppression de bits inutiles)
- \* Localisation de l'erreur (par exemple, en cas d'utilisation de code auto-correcteur d'erreur)
- \* Contrôle d'alignement, contrôle de la longueur de la trame, contrôle du débordement par sur-vitesse, autres types de contrôle

. Gestion des services

- \* Décryptage, autres ...

Dans notre analyse, le processus Accès à la Mémoire prend en compte la gestion des différentes structures de données (vecteur, puits, liste, anneau, autres) en mémoire externe, qui auront été préparées par le niveau LLC ou équivalent (que le niveau LLC soit implanté sous forme de logiciel et de système d'exploitation tournant dans un microprocesseur, ou qu'il soit implanté sous forme de machine dédiée ou câblée). Ce processus est souvent appelé DMA dans différentes implantations. Le processus Tampon permet l'adaptation du débit du bus série à celui du bus parallèle [DAN 87.2].

Dans le cas où les informations de commandes se trouvent dans la mémoire externe, le processus qui les traite est un autre processus Accès à la Mémoire et il faut gérer le bus de communication avec l'Interface Station. Si le circuit contrôleur est directement adressable par la station, ces informations de commandes peuvent aussi se trouver dans la mémoire interne du circuit contrôleur.

Les processus qui composent le processus Emission de Trame sont deux de ceux présentés pour le processus Réception.

+ Le processus *Calcul des Statistiques* doit gérer :

- . les informations statistiques courantes dont il est nécessaire de doter les circuits de communication pour le suivi dynamique souhaité par l'utilisateur (par exemple : nombre de collisions, fréquence de retransmission, etc ...),
- . les différentes mesures spécifiques à la méthode d'accès (par exemple les mesures de réflectométrie dans le cas de l'IEEE 802.3),
- . le compte-rendu de ces informations.

+ Le processus *Initialisation* doit gérer :

- . la lecture/écriture des paramètres du circuit,
- . la reconfiguration de l'ensemble du circuit après "reset".

+ Le processus *Service* doit gérer

- . la définition de la base de temps,
- . les différents types de test dont la sémantique est liée à la méthode d'accès et plus généralement à la communication (les tests miroir par exemple).

### 2.323. CARACTERISTIQUES DU MODELE PROPOSE

On observe qu'il y a principalement deux flots de données : une à l'émission et une à la réception, sachant que le flot de données en réception est prioritaire.

Le seul objet dynamiquement géré dans le circuit est la trame que le circuit est en train d'émettre ou que le circuit est en train de recevoir. Si l'on assume que le temps de traitement d'une trame en réception (resp. en émission) est inférieur à la borne inférieure de l'intervalle de temps inter-frames, il n'est effectivement pas nécessaire de prévoir des processus de réception (respectivement émission) *dynamiques* pour traiter les différentes trames de données. L'existence de ces processus peut donc être prévue *statiquement* dans la définition de l'architecture fonctionnelle du circuit contrôleur.

Le modèle d'architecture fonctionnelle que nous proposons pour cette classe de circuits contrôleur de communication du niveau MAC des réseaux locaux est celui schématisé ci-dessus, avec un seul processus Réception de Trame et un seul processus Emission de Trame. Ce réseau d'unités asynchrones a les caractéristiques suivantes :

- . Chaque unité exécute un processus déterminé.
- . Le nombre d'unités est statiquement défini à la conception.
- . Les unités peuvent a priori partager de la mémoire ou des ressources communes.

. Les deux types de communication d'informations entre les unités existent : fortement couplée dans le cas des communications synchrones entre deux unités qui partagent une horloge commune, et faiblement couplée entre unités asynchrones.

. Le contrôle du fonctionnement du modèle est généralement de type maître-esclave : un processus maître dirige l'ensemble des processus esclaves.

Note : Le partage de ressources communes (bus parallèle, mémoire, processus maître) et l'existence des deux flots de données font que le parallélisme réel entre certaines unités asynchrones de ce modèle est plutôt une exception que la règle générale d'implantation.

Ce modèle d'architecture ainsi que celui présenté pour les méthodes d'accès sur des bus multi-microprocesseurs peuvent être utilisés, par raffinement successif, pour la définition de l'architecture fonctionnelle de chaque unité. L'exemple de décomposition du processus Réception de Trame est caractéristique de cette approche.

## 2.33. Travaux réalisés

### 2.331. LE CIRCUIT FIP-VLSI

#### Introduction

FIP est un projet français qui émane d'une initiative du Ministère de la Recherche et de la Technologie, en 1984. C'est un système de transmission série multiplexée pour les échanges d'informations entre des capteurs, des actionneurs et des automates réflexes. Nous avons participé à ce projet à divers titres :

- . d'abord, en tant qu'expert dans la rédaction du Livre Blanc sur les caractéristiques générales du système de communication FIP [GAL 84];
- . ensuite comme animateur du groupe FIP-Capteurs dont le rapport de synthèse [DAN 86.2] vient d'être présenté, après une longue période de discussion, à la CIAME (Commission Industrie-Administration pour la MESure - groupe de travail sur les Capteurs Intelligents) et
- . enfin, en tant chercheur dans la mesure où nous avons proposé, en avance de phase par rapport au travail des industriels, les spécifications d'un circuit intégré à haute densité d'intégration, appelé FIP-VLSI, destiné à connecter des capteurs simples ou intelligents au bus FIP [DIA 86.2].

La conception de ce circuit de recherche a été menée dans notre équipe avec l'aide du CNET CNS [DIA 86.1].

#### Architecture fonctionnelle

La conception du circuit FIP-VLSI a été précédée par l'étude de son architecture fonctionnelle, selon le modèle présenté dans le paragraphe précédent. Il intègre les différents processus suivants : Interface avec la Station, Initialisation, Gestion de la Trame, Interface avec le Medium et Gestion des Services [DAN 86.4].

Parmi ces processus, le processus Initialisation et le processus Interface avec le Medium comportent des choix système particuliers qui méritent d'être soulignés. Parmi les fonctions gérées par le processus Initialisation, signalons :

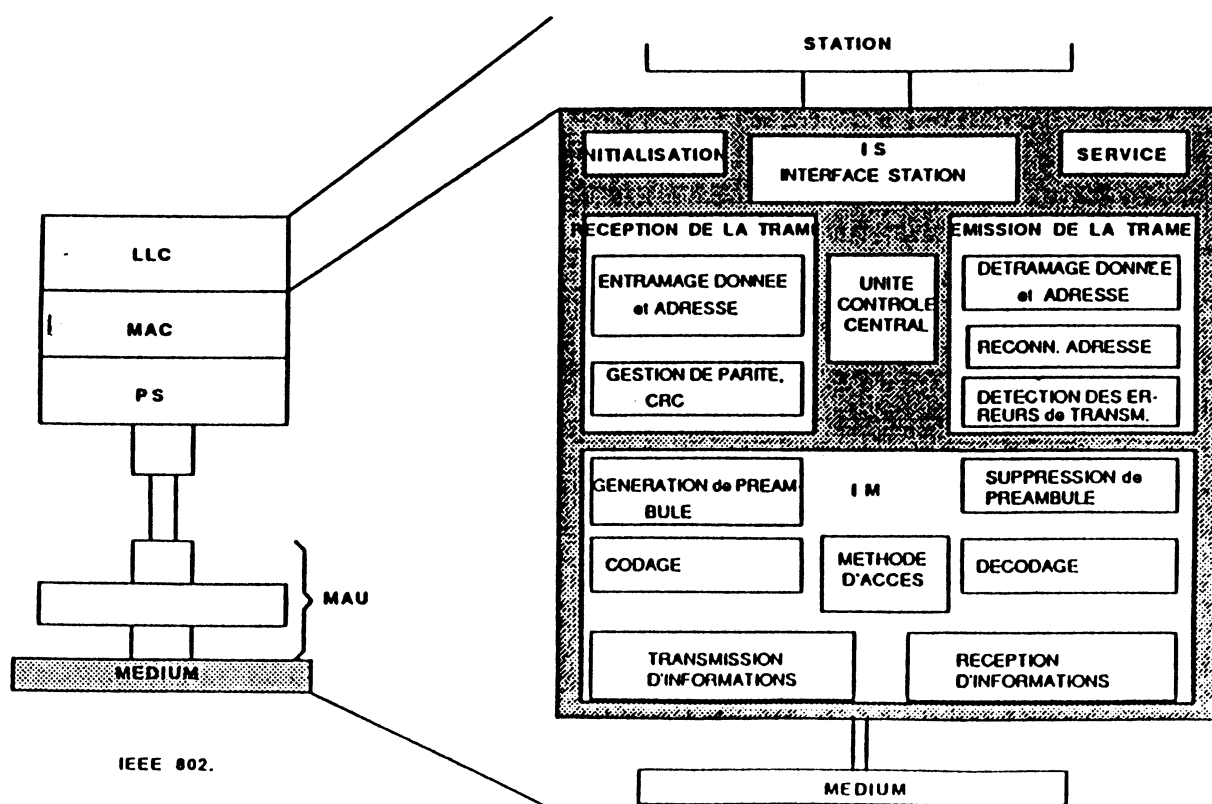
- . *le chargement statique de l'adresse de la station dans le circuit* (cette méthode de chargement de l'adresse à l'initialisation constitue un apport original au débat sur la méthode de chargement statique de l'adresse d'un couple 'station non intelligente - circuit de communication', sur sa localisation et sur la méthode de reconnaissance de l'adresse ), et

- . *la définition statique de la taille des chemins de données utilisées à l'interface entre la station et le circuit* (autre apport original dans la gestion souple de l'interface entre une station non

intelligente et le circuit de communication ).

D'autre part, le circuit FIP-VLSI incorpore dans son Interface avec le Medium des fonctionnalités qui font généralement partie des outils de ligne, telles que la génération (ou la reconnaissance) du préambule et du motif de synchronisation, le codage (ou le décodage) de la trame. Ce choix a été en fait dicté par l'absence de composant 'outils de ligne' dans le projet FIP, car notre conception du circuit FIP-VLSI était en avance de phase par rapport au projet lui-même. Il représentait d'autre part une opportunité de développer notre savoir-faire en ce qui concerne le codage et le décodage du code Manchester.

L'architecture fonctionnelle du circuit FIP-VLSI est schématisée dans la figure suivante :



### Méthode de synthèse

La méthode de synthèse que nous préconisons consiste à synthétiser processus (ou unité asynchrone) par processus selon les fonctions qui composent chaque processus. Différents types de synthèse cible, parmi ceux présentés dans le paragraphe 2.23, sont utilisés dans le cas du circuit FIP-VLSI :



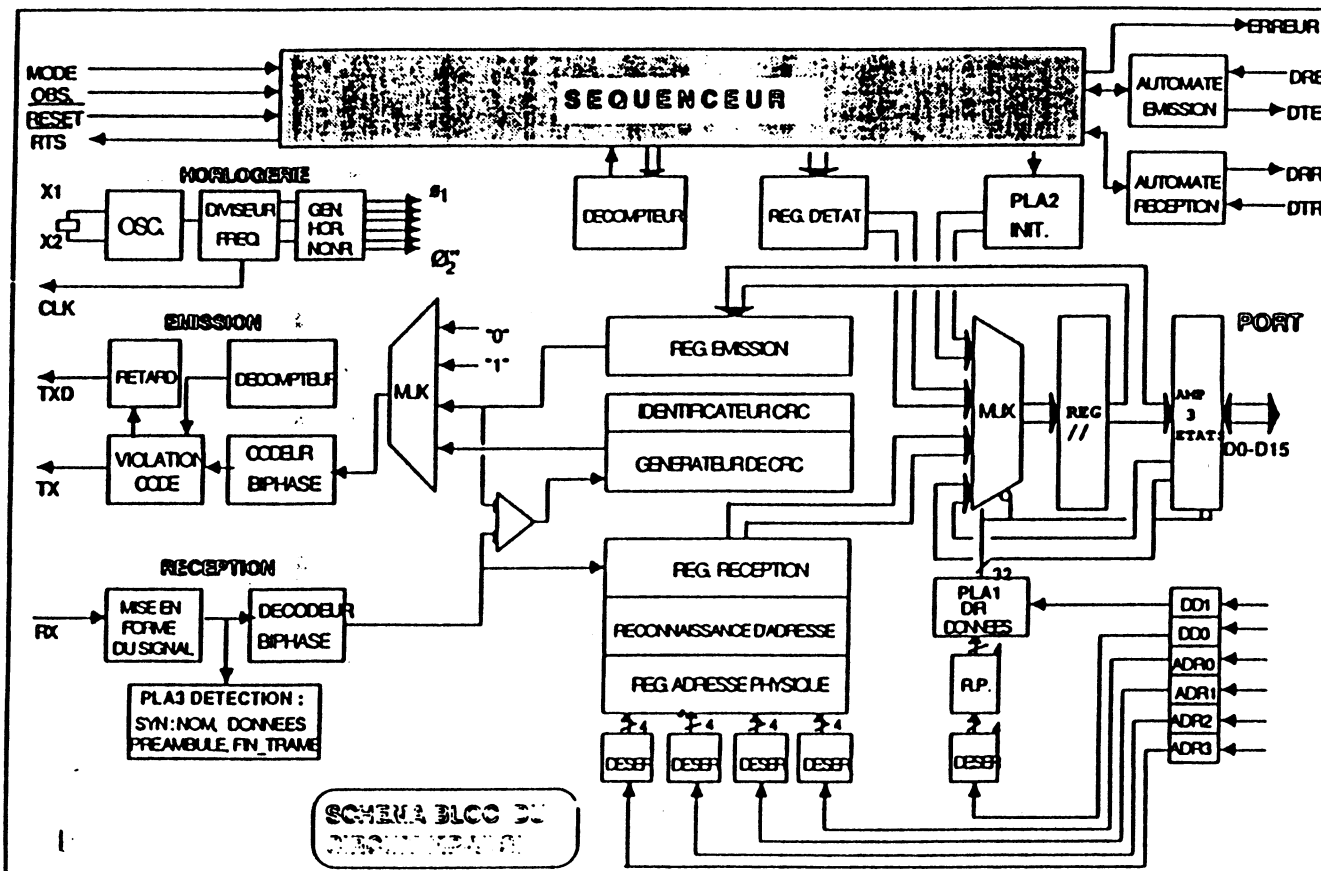
- . synthèse des parties opératives en tranches : registre à décalage, différents types de désérialisateurs, sérialisateur, registre parallèle, différents types de compteurs, générateur de CRC;
- . synthèse en logique aléatoire (réseau de portes de base ou complexes) : notamment pour des problèmes de violation de code dans l'Interface Medium, pour des automates d'états finis;
- . synthèse de partie contrôle sous forme de séquenceur : partie contrôle central du circuit, chargement de l'adresse à l'initialisation.

La synthèse s'effectue par raffinement successif, processus par processus et fonction par fonction. Le tableau suivant montre le résultat de cette étude de synthèse.

PROCESSUS		FONCTIONS		OPERATEURS
INTERFACE STATION		TRANSFERT DES DONNEES	PORT	REGISTRE PARALLELE
		SYNCHRONISA- TION DU TRANS- FERT	EMISSION	AUTOMATE D'ETATS FINIS
			RECEPTION	AUTOMATE D'ETATS FINIS
INITIALI- SATION		DEF. DE LA TAR- LE DU CHEMIN DE DONNEES		ENCODEUR
		DIRECTION DES DONNEES		REGISTRE DE MEMORISATION
		CHARGEMENT DE L'ADRESSE	AD. PHYSIQUE	REGISTRE / REGISTRE A DECAL DECODEUR
GESTION DE LA TRAME	EMISSION DE LA TRAME	ENTRAMAGE		SERIALISATEUR
		GENERATION CRC		GENEREUR CRC
	RECEPTION DE LA TRAME	RECONNAISSAN- CE DE L'ADRES- SE		REGISTRE / REGISTRE A DECAL COMPARATEUR
		VERIFICATION CRC		GENEREUR CRC COMPARATEUR
		DETRAMAGE		DESERIALISATEUR
INTERFACE MEDIUM	METHODE D'ACCES			SEQUENCEUR
	EMISSION	GENERATION DE PREAMBULE		COMPTEUR
		CODAGE	CODEUR BIPHASE VIOLATION DU CODE	LOG. ALEATOIRE DECOMPTEUR LOG. ALEATOIRE
	RECEPTION	REMISE EN FORME		LOG. ALEATOIRE
		SUPPRESSION DE PREAMBULE		DESERIALISATEUR DECODEUR
		DECODAGE		REG. A DECALAGE LOGIQUE ALEATOIRE
SERVICE		GENEREUR D'HORLOGES		OSCILLATEUR
				DMISEUR DE FREQ.

Analyse fonctionnelle du circuit FIP-VLSI

L'architecture en terme de blocs d'opérateurs du circuit FIP-VLSI est présentée dans la figure suivante :



## Conclusion

Développé dans le cadre des travaux qui sont connexes au projet FIP sur le plan national français, *en avance de phase et de fonctionnalités* par rapport aux travaux industriels (circuit FIP automate en prédiffusé par la CSEE [ROD 86] ),

la conception de la maquette du circuit FIP-VLSI vise à démontrer la faisabilité d'un certain nombre de propositions concernant l'interface pour les stations capteurs simples, l'initialisation du composant et la méthode d'adressage.

Elle a permis d'autre part la mise au point d'une méthodologie de conception de circuits VLSI destinés à la communication pour les niveaux bas du modèle ISO. Comme cette étude a été menée jusqu'à l'implantation du circuit, sa conclusion se trouve déportée en partie dans le paragraphe § 4. 21.

## 2.332. ETUDE DE LA COMPLEXITE ET DE LA FAISABILITE DU CIRCUIT CSMA/DCR

### Introduction

Dans le cadre de nos activités de recherche en collaboration avec d'autres équipes de recherche, nous avons effectué pour le compte du projet SCORE de l'INRIA [BOU 86] l'étude de la complexité du circuit CSMA/DCR et de la faisabilité de son implantation en technologie CMOS disponible dans l'industrie française [DAN 86.3].

Le futur circuit CSMA/DCR sera un circuit de niveau MAC, qui implante une méthode d'accès à base de contention CSMA [SHO 80] dont l'algorithme suivi pour la retransmission après collision est un algorithme déterministe, basé sur un parcours de l'arbre binaire constitué par les adresses physiques des contrôleurs de communication [ROL 87] [BOU 86]. Cet algorithme est dérivé de travaux déjà connus [CAP 79] mais les a notablement améliorés, en particulier en ce qui concerne la garantie d'un temps fini et borné supérieurement pour la transmission d'un événement ou d'un message.

### Etude de la complexité et de la faisabilité du circuit

Cette étude a nécessité la spécification de l'architecture fonctionnelle du circuit, selon le modèle présenté dans le paragraphe 2.32. En suivant la démarche préconisée, nous avons été amenés à :

- . à étudier l'architecture de chaque processus et de chaque composant du processus,
- . à faire la synthèse logique (définir le schéma logique de tout module câblé) pour pouvoir en évaluer la complexité ,
- . à faire des hypothèses sur (et par conséquent à recommander) une architecture matérielle du futur circuit et
- . à générer, dans les cas complexes, l'architecture implantée pour avoir une première évaluation approximative de la surface.

Pour illustrer la démarche, nous allons seulement décrire le processus Réception d'une Trame, qui est l'un des plus complexes du circuit.

A. Le processus Réception d'une Trame se décompose en quatre processus principaux.

- . Le processus Communication avec l'Interface Medium, appelé dans la figure suivante module Réception.

- . Le processus Traitement en temps-réel des éléments du protocole qui comprend :
  - + le module Reconnaissance d'adresse,
  - + le module CRC, (le module CRC fait partie dans ce processus dans la mesure où il est utilisé pour la reconnaissance des adresses de groupe)

+ le module Traitement de l'horloge réseau.

. Le processus Transfert des informations qui comprend :

+ la gestion de la synchronisation avec l'Interface Station,

+ le module DMA de réception d'informations,

+ le module FIFO.

. Le processus Traitement des commandes de la station qui comprend :

+ la gestion de la synchronisation avec l'Interface Station,

+ le module DMA de réception des commandes,

+ le traitement des commandes (modification dynamique des adresses, ...)

B. La synthèse logique du processus Réception d'une Trame est illustrée par le tableau suivant.

PROCESSUS		FONCTIONS		OPERATEURS
RECEPTION DE LA TRAME	INTERFACE AVEC L'IM	RECEPTION	DETRAMAGE	DESERIALISATEUR
			BARRIERE TEMPORELLE	REGISTRE + logique de contrôle
			DETECTION PREAMBULE	AUTOMATE D'ETATS FINIS
				COMPTEURS REGISTRES
			partie contrôle	SEQUENCEUR
	TRAITEMENT EN TEMPS-REEL DES ELEMENTS DU PROTOCOLE	RECONNAISSANCE DE L'ADRESSE	ADRESSE PHYSIQUE ADRESSE DIFFUSION ADRESSE GROUPE	REGISTRES RAM + logique de contrôle
		VERIFICATION CRC		GENERA TEUR CRC COMPARATEUR + logique de contrôle
		TRAITEMENT DE L'HORLOGE RESEAU		COMPTEURS REGISTRES + logique de contrôle
	TRANSFERT DES INFORMATIONS VERS LA STATION	DMA DE RECEPTION DES INFORMATIONS		REGISTRES COMPTEURS DECOMPTEURS
			partie contrôle	SEQUENCEUR
		FIFO		COMPTEURS RAM + logique de contrôle
	TRAITEMENT DES COMMANDES DE LA STATION	DMA DE RECEPTION DES COMMANDES		REGISTRES COMPTEURS DECOMPTEURS
partie contrôle			SEQUENCEUR	
UNITE DE CONTROLE DE RECEPTION	partie contrôle		SEQUENCEUR	

Analyse fonctionnelle et synthèse logique du processus Réception d'une Trame

C. L'architecture de chaque fonction ou module analysé est ensuite décrite en terme de partie

opérative et partie contrôle. Nous avons pris pour hypothèses d'évaluer ces parties opératives et ces parties contrôle en tant que structures régulières (bâties à partir des bibliothèques de cellules et à partir des outils de génération automatique de partie contrôle du CNET CNS).

Il en est de même pour l'Unité de Contrôle de Réception qui est le processus maître qui contrôle le fonctionnement de la machine Réception d'une Trame.

### Résultats

Le détail complet de cette étude se trouve dans les rapports [DAN 86.3].

. Le premier rapport concerne la définition de l'architecture du circuit et l'étude de la partie opérative de chaque module obtenu dans la décomposition.

. Le deuxième rapport concerne l'étude de la partie contrôle de chaque module. Chaque partie contrôle est décrite en GRAFCET. A partir de chaque description, une ou plusieurs architectures cible sont générées (en mono-PLA dans les cas simples; dans les cas complexes, deux architectures cible sont proposées et comparées : en mono-PLA ou en ROM de microcommandes et PLA de contrôle). Les masques correspondants sont générés pour permettre une première évaluation approximative de surface.

L'évaluation de la complexité du circuit CSMA-DCR a été faite à partir d'une bibliothèque de cellules disponibles dans le projet CAO système SCHUSS du CNET CNS avec des outils de CAO spécifiques tels que le générateur de PLA, le générateur de RAM. La technologie de référence pour cette étude est une technologie CMOS 2  $\mu$  et 1 métal.

La complexité du circuit est estimée à plus de 100.000 transistors. Nous avons estimé, en 1986 lors de l'étude, que ce circuit est faisable en une seule puce; l'évolution passée de la technologie CMOS vers un pas de grille de l'ordre de 1,25  $\mu$  et vers l'utilisation de deux métaux d'interconnexion ne fait que nous conforter dans ce choix.

### Conclusion

Ces travaux nous ont donc permis de mener à l'université une étude de faisabilité technologique et de complexité d'un circuit de communication qui a fait, depuis, l'objet d'une conception et d'une implantation effectuées par ESD sous la forme de trois circuits différents. Il est vrai que cette conception a dû intégrer des contraintes additionnelles qui n'avaient pas été présentes dans notre cahier des charges.

Ils ont permis d'autre part d'effectuer l'étude et la réalisation d'un générateur de bloc flexible qui gère l'accès direct à une mémoire structurée en liste [DAN 88.3]. Comme l'étude concerne notre méthodologie de conception, elle fait l'objet d'un développement dans le chapitre 4.3.

## 2.34. Recommandations pour les choix d'architecture matérielle

L'approche que nous préconisons pour choisir l'architecture matérielle des circuits de communication du niveau MAC est une approche multi-machines câblées, par opposition à l'approche processeur parallèle à usage général.

Selon une démarche hiérarchique et modulaire, l'étude par raffinement successif du circuit passe par l'étude de chaque unité asynchrone. L'architecture matérielle de chaque unité asynchrone peut être étudiée dans un premier temps de manière indépendante.

Chaque machine câblée implantant une unité asynchrone sera définie sous la forme d'une partie contrôle et d'une partie opérative; la partie opérative d'une machine peut contenir plusieurs éléments, chacun d'entre eux pouvant être implanté de nouveau comme une machine câblée.

Ainsi, l'architecture d'une machine se bâtit-elle par raffinements successifs, ponctués par des retours en arrière pour tenir compte de multiples contraintes système, d'interface, de facilité de test et de faisabilité technologique.

On remarque que la partie contrôle central du circuit de communication, ainsi que celle de certaines unités sont d'une grande complexité. Cependant, le savoir-faire [OBR 82] et des outils sophistiqués de conception concernant l'implantation des parties contrôle existent. Ceci est dû à un héritage datant des premières conceptions des circuits microprocesseur qui a fructifié avec le temps. Différentes architectures matérielles existent (§ 2.23), sont à génération automatique et de ce fait facilement simulables (ce qui permet une évaluation relativement précise des performances, des contraintes de forme et de surface).

En ce qui concerne les parties opératives à architecture régulière, nous préconisons une approche de type 'tranche de bit' [MEA 81] avec la prise en compte d'une stratégie de routage intercellules et interblocs [ROU 86][DAN 88.4].

En conclusion, l'approche multi-machines câblées est en fait une approche pratique, mais elle manque d'évidence de souplesse. Nous avons choisi de lui donner un nom : nous appelons ce type d'architecture une **architecture semi-dédiée**. Elle est dédiée car elle est faite pour un domaine système donné. L'expérience des conceptions passées a légué des outils créés pour ce faire (bibliothèques de cellules, bibliothèques d'opérateurs, générateurs de blocs flexibles, solutions et algorithmes trouvés pour résoudre des problèmes généralisables, outils de CAO spécifiques). Cela dit, un circuit doit être conçu de nouveau pour implanter le nouveau protocole de communication, même si la définition de l'architecture de ce nouveau circuit peut utiliser le modèle proposé, les connaissances et les méthodes afférentes ainsi que les outils existants.

Selon l'analyse que nous avons faite :

. de l'ensemble des protocoles de communication du niveau MAC (de leur complexité et des différences sémantiques notables entre leurs modes de fonctionnement),

. de la majorité des circuits existants qui implantent ces protocoles (qui ont tous adopté l'approche multi-machines câblées - mis à part Texas Instruments pour *les* circuits qui implantent le protocole IEEE 802.5, jeton sur anneau d'IBM [MOK 84]) et

. de nos propres implantations et études, FIP-VLSI et CSMA-DCR

nous affirmons que *l'approche processeur à architecture parallèle et à usage général* pour gérer la communication du niveau MAC du modèle IEEE n'est pas satisfaisante. Cela principalement pour des raisons de performances à atteindre dans certains cas très précis (la reconnaissance d'une adresse, le calcul d'un CRC, la reconnaissance d'une collision faite par calcul du CRC) et pour des raisons de coût (qui dit processeur d'usage général dit choix de bus mémoire, dit composants mémoires compatibles).

A contrario, cette approche est plus adaptée, comme nous allons le voir, à l'implantation des protocoles des couches hautes du modèle OSI.

## Chapitre 3

# ARCHITECTURE DE CIRCUITS DE COMMUNICATION POUR LES COUCHES 3, 4, 5 ISO



### 3.1. INTRODUCTION

L'architecture des circuits intégrés qui implantent les couches hautes de l'OSI ou les couches équivalentes est un sujet de recherche relativement neuf. Par conséquent, il faut considérer le développement qui suit comme une réflexion sur ce sujet.

Cette réflexion se limite aux couches qui sont indépendantes des applications : Réseau, Transport et Session. En ce qui concerne les couches Présentation et Application, il faut attendre que les instances internationales de normalisation puissent figer leurs travaux et produire des spécifications durables. Elles fourniront des sujets de recherche en architecture de circuits pour les années à venir.

Dans le développement suivant, nous considérons que le lecteur connaît déjà l'architecture des Interconnexions des Systèmes Ouverts OSI de l'ISO [ISO 84] ainsi que les fonctionnalités des couches Réseau, Transport et Session.

Une première approche pratique du problème de l'implantation matérielle des couches hautes du modèle OSI consiste à développer du logiciel destiné à être figé sur des cartes; ces cartes seront développées pour un environnement microprocesseur donné, composé d'une famille de circuits intégrés compatibles et d'un bus fond-de-panier donné. Cette approche est généralement celle des industriels du domaine de la communication des données. Citons les produits d'OST X25.3 [OST 87] bâtis avec le bus VME, les produits logiciels du réseau local industriel FACTOR [APT 87.1], les produits INA 960 d'INTEL basés sur la carte MULTIBUS ;186/51 [INT 84.1], la carte X25.3 de THOMSON [THO 87]. C'est dans cette approche que les principaux efforts sont généralement fournis. Ils ont pour but l'amélioration de la conception du logiciel ou de la réalisation des cartes afin de les rendre plus modulaires et plus souples d'emploi; l'aspect architecture des cartes coupleur de connexion ou des circuits contrôleur de communication est comparativement moins étudié.

Une deuxième approche du problème de l'implantation matérielle des couches hautes consiste à considérer que les connaissances acquises dans le domaine de la conception des circuits de communication du niveau MAC (INTEL 82586, MOTOROLA 68824) ou LLC (WD 2501, WD 2511.A, ...) peuvent être exportées et serviront de base à la conception des circuits implantant les couches hautes du modèle OSI. Pour des raisons qui vont apparaître clairement dans l'exposé qui suit (parallélisme implicite entre langage applicatif et système d'exploitation, parallélisme entre deux couches adjacentes, nouvelles fonctions telles que les fonctions de bout en bout, objets dynamiques tels que les connexions logiques, ...), cette deuxième approche risque de conduire à quelques impasses.

Notre réflexion se situe à mi-chemin entre ces deux approches, mais dans le cadre plus général d'une machine à architecture parallèle et à usage général pour le traitement des protocoles de communication digitale. Elle a été effectuée initialement dans un contexte de recherche et

développement industriel pour le compte d'un travail d'étude sur *la spécification fonctionnelle et la définition architecturale d'un composant baptisé MC3, MicroContrôleur de Communication Configurable, destiné à implanter séparément ou conjointement les couches 3, 4, 5 de l'ISO ou équivalentes : c'est-à-dire précisément Réseau (X25 niveau Paquet), Transport (jusqu'à la classe 4) et Session (BCS et BAS) [DAN 87.3] [DAN 87.4].*

L'avant projet d'étude concernant sa faisabilité a été effectué mi-1985 [DAN 85.2]. Une présentation générale du projet et des premiers résultats se trouve dans [ANS 88].

### **3.12. Le projet MC3**

Le circuit MC3 doit implanter individuellement soit chaque couche annoncée, soit un des trois sous-ensembles de couches suivants : (3, 4), (4, 5), (3, 4, 5). Dans chaque hypothèse, une borne supérieure pour le nombre de connexions logiques autorisées et une borne inférieure pour le débit sont définies avec l'administration de la Direction Générale des Télécommunications, initiatrice de ce projet.

Le circuit MC3 doit pouvoir à terme être utilisé dans la connexion des terminaux et des ordinateurs aux réseaux généraux X25, publics ou privés. Cette contrainte explique le choix de la couche X25 niveau Paquet du CCITT pour l'étude d'architecture et de sa validation.

Il doit pouvoir être utilisé dans des stations connectées à des réseaux locaux. Cette contrainte explique le choix de la classe 4 de la couche Transport. Mentionnons en particulier que le projet MAP [KAM 86][ESP 86] recommande l'implantation de la même classe pour le protocole de Transport.

Il doit pouvoir être, à notre avis, utilisé pour faire l'interface de communication des stations de travail à hautes performances. De ce fait, nous ne pouvons pas exclure la possibilité d'implanter une couche de Transport, type TCP-IP, sur MC3.

Toutes les fonctionnalités qui sont recommandées par les comités internationaux de standardisation (ISO, CCITT, MAP, ...) doivent être prises en considération par l'étude de l'architecture de MC3. En particulier, la possibilité pour MC3 de réaliser la segmentation, la concaténation et le multiplexage à l'émission à travers les différentes couches implantées (respectivement le réassemblage, la séparation et le démultiplexage). Il faut rappeler que ces fonctionnalités sont propres à chaque couche et que leur sémantique diffère d'une couche à l'autre. Le rappel de cette hypothèse de travail peut sembler superflu, encore faut-il mentionner que ces fonctionnalités sont souvent ignorées dans la plupart des implantations industrielles.

## 3.2. ARCHITECTURE FONCTIONNELLE DU CIRCUIT DE COMMUNICATION MC3

### 3.2.1. Analyse de synthèse des implantations logicielles et matérielles existant

L'ensemble des implantations des protocoles des couches hautes ISO, CCITT ou de niveau équivalent que nous avons choisi d'analyser, se décompose en deux sous-ensembles différents :

+ Le premier groupe concerne des implantations logicielles des couches Transport et Session dans un contexte où la visibilité de l'environnement matériel est réduite, sinon voisine de l'absence de visibilité. Ce sont les implantations suivantes :

. Implantation A de la Station de Transport version 2 sur IBM 360/67 [DAN 76]. Le réseau de communication est le réseau général à commutation de paquets Cigale du réseau Cyclades [POU 82].

. Implantation B de la couche Transport ISO et de la couche Session ISO : il s'agit d'un produit industriel portable [MAR 87] qui est écrit en langage C et est présenté comme indépendant du système d'exploitation hôte.

+ Le deuxième groupe concerne les implantations des couches Session et/ou Transport et/ou Réseau dans un contexte où la visibilité de l'environnement matériel est importante (environnement microprocesseur). Ces logiciels implantés sont généralement destinés à être chargeables en mémoire volatile ou à être figés en mémoire morte. L'analyse porte sur les implantations en "firmware" suivantes :

. Implantation C d'un logiciel Transport et d'un logiciel Session pour le réseau local industriel Factor [FAC 87]. Cette implantation est faite dans un environnement basé sur la famille des composants INTEL compatibles avec le microprocesseur 8086.

. Implantations logicielles et matérielles de la couche X25.3. Les deux implantations analysées sont les suivantes :

- Implantation D d'une couche X25.3, 2 et 1 pour l'évaluation des protocoles ARCHITEL [ARC 86] , réalisée par le CCETT [HEN 85].

- Implantation E des couches X25.3, 2 et 1 pour permettre la connexion des équipements aux réseaux X25 (publics comme TRANSPAC ou privés comme TYMNET, etc ...). Il s'agit d'une gamme de produits industriels [OST 87] développés pour différentes architectures fond-de-panier : VME, bus IBM PC XT, bus IBM PC AT, ... .

Les difficultés inhérentes à ce genre d'analyse sont évidentes, surtout en ce qui concerne l'analyse ou l'observation des performances. Les données manquantes sont, comme il est d'usage, à mettre au passif de celui qui analyse. Les données présentées sont, comme il est toujours d'usage, sous la seule responsabilité du rédacteur.

Cependant, comme nous l'avons fait pour analyser les bus parallèles et 'série', nous avons adopté une méthode uniforme qui est basée en premier lieu sur une analyse en terme de flots de données des différentes implantations, compte tenu de leur spécificité propre.

La synthèse de l'analyse peut seulement être effectuée à partir du moment où l'on est apte à identifier les fonctionnalités et les sous-systèmes principaux de toutes les implantations.

Une présentation générale de cette synthèse de l'analyse se trouve dans [DAN 88.2].

### 3.211. FONCTIONNALITES IDENTIFIEES

Les fonctionnalités identifiées sont regroupées sous les rubriques suivantes :

+ *Gestion du Parallélisme* : Les critères de décomposition du problème en processus concurrents et communicants ont été répertoriés. La gestion des processus (type de processus, méthode d'allocation, définition de la priorité entre processus, définition de la préemptibilité des processus, ré-entrée des programmes, présence d'un noyau système temps-réel), la gestion de la communication et de la synchronisation entre processus, la gestion du partage des ressources ont été analysées.

+ *Fonctionnalités directement dépendantes de l'environnement matériel* : La gestion de la mémoire (perçue par toutes les implantations comme fonctionnalité prioritaire, la gestion mémoire a fait l'objet de plusieurs approches différentes), les différents types des données et leur représentation, la gestion des interruptions et la gestion des chiens de garde ont été analysés.

### 3.212. SOUS-SYSTEMES PRINCIPAUX

Les principaux sous-systèmes suivants ont toujours été observés dans toutes les implantations : le *noyau système* (quelle soit l'importance qu'il peut avoir), le *traitement des interfaces* haute et basse et le *traitement du protocole* de communication proprement dit.

Après l'identification de ces sous-systèmes principaux dont nous considérons l'existence comme postulat ou à défaut comme assertion dans notre travail, nous allons mener l'investigation de nos propres implantations, en l'occurrence A et C, pour évaluer leur importance respective. Les résultats de cette évaluation serviront de base de déduction pour définir l'architecture du composant MC3.

## 3.213. SYNTHÈSE

Soit les hypothèses de travail suivantes :

*Hyp1* : Le protocole observé est un protocole de Transport type classe 4 (cas A) qui fonctionne avec une seule connexion, soit à l'émission, soit à la réception.

*Hyp2* : Le protocole observé est un protocole de Transport type classe 4 (cas A) qui fonctionne avec 5 connexions, sachant que sur chaque connexion, il y a une anticipation possible de 3 TPDU's.

*Hyp3* : Le terme *coût* utilisé ci-après est défini, selon une première approximation, à partir du nombre d'instructions assembleur exécutées.

*Hyp4* : Le terme noyau système englobe les fonctionnalités suivantes : commutation et gestion des processus, gestion de la communication et de la synchronisation entre processus, gestion de la mémoire, gestion des chiens de garde, gestion des interruptions internes et externes et gestion intégrée du débogage.

Nous avons mesuré l'importance comparative des sous-systèmes annoncés en mesurant les rapports suivants :

$$R1 = \frac{\text{Coût du traitement de l'interface (5,4)} \\ \text{(y compris le traitement correspondant dans le noyau)}}{\text{Coût du traitement du protocole correspondant} \\ \text{(y compris le traitement correspondant dans le noyau)}}$$

$$R2 = \frac{\text{Coût du traitement du protocole hors noyau}}{\text{Coût du traitement correspondant dans le noyau}}$$

$$R3 = \frac{\text{Coût du traitement de l'interface (5,4) hors noyau}}{\text{Coût du traitement correspondant dans le noyau}}$$

Les résultats suivants ont été obtenus dans le cas des implantations A et C :

		<i>emission</i>	<i>reception</i>		
		A		C	
R 1	hyp. 1	0,93	0,99	2,13	le coût du traitement de l'interface est du
	hyp. 2	0,87	0,93		même ordre de grandeur ou supérieur au
					coût du trait. du protocole correspondant
R 2	hyp. 1	1	0,89	1,46	le coût de traitement du protocole est du
	hyp. 2	1,46	1,17		même ordre de grandeur que le coût du tra-
					tement correspondant dans le noyau syst.
R 3	hyp. 1	0,52	0,55	0,25	le coût du traitement de l'interface est
	hyp. 2	0,75	0,68		inférieur au coût du traitement correspon-
					dant dans le noyau système

Des mesures concernant le protocole de la couche Session ont d'autre part été effectuées dans le cas de l'implantation C. Elles ont fourni des résultats du même ordre de grandeur.

### 3.214. PROPOSITIONS POUR LA DEFINITION DE L'ARCHITECTURE FONCTIONNELLE DE MC3

Ces résultats et des observations complémentaires nous amènent à faire les propositions générales suivantes concernant les points qui doivent être considérés comme critiques dans la définition de l'architecture d'un composant tel que MC3 ou d'un système matériel visant l'implantation des protocoles des couches hautes telles que Session, Transport ou Réseau. Ces propositions représentent en fait le cahier des charges générales pour cette définition.

**P1.** Le parallélisme entre le traitement du protocole, le traitement de l'interface et le noyau système doit être aussi efficace que possible.

**P2.** Le traitement de l'interface doit être conçu :

a. de manière spécifique car les fonctionnalités d'une interface donnée sont définies par un standard et ont une sémantique propre aux couches qui communiquent via cette interface, et paradoxalement

b. de manière à ce qu'elle soit indépendante de la couche traitée, pour d'évidentes raisons de modularité.

c. L'interface entre deux couches adjacentes traitées par un seul composant est une interface interne. Dans ce cas, le traitement de l'interface doit pouvoir être dégradé (allégé) car l'environnement de cette interface est homogène et connu.

**P3.** Dans les implantations analysées, les fonctionnalités suivantes sont très coûteuses en temps d'exécution, et de ce fait nécessitent une implantation matérielle optimisée :

- a. la gestion mémoire (en conjonction avec le traitement de l'interface),
- b. la gestion de la communication entre processus et
- c. la gestion du basculement de contexte entre processus.

Certaines de ces propositions ont aussi été mises en exergue dans d'autres implantations, logicielles ou matérielles. Mentionnons :

. Les mesures de performance d'un protocole HDLC concernant les instructions de manipulation de la mémoire [TOB 87]; elles sont à prendre en compte dans le cadre des propositions P2.a et P3.a.

. La notion de rendez-vous dans la communication entre processus [HOA 79], qui est implantée dans les circuits microprocesseur INMOS TRANSPUTER; elle est à prendre en compte dans le cadre de la proposition P3.b.

. La gestion optimisée du basculement de contexte dans les plus récents circuits microprocesseur, tels que l'INTEL 80386, l'INMOS TRANSPUTER; elle est à prendre en compte dans le cadre de la proposition P3.c.

### 3.215. EVALUATION DE LA COMPLEXITE D'UN ALGORITHME D'EMISSION OU DE LA RECEPTION D'UNE TRAME

L'évaluation de la complexité des implantations A et C a été faite avec les hypothèses suivantes :

. Elle est évaluée, à la première approximation, en terme de nombre d'instructions assembleur exécutées (des trois sous-systèmes principaux donnés : traitement de l'interface, traitement du protocole et noyau système) pour émettre ou pour recevoir un paquet d'une taille N donnée .

. Nous supposons qu'il n'y a pas de problèmes d'attente aux interfaces avec les couches adjacentes.

. Les hypothèses Hyp1 et Hyp2, définies ci-dessus, sont maintenues.

Les résultats suivants ont été obtenus :

	A		C	cette implantation (C) est écrite en PL86; le prog. en assembleur est obtenu par génération, donc non optimisé
emission	hyp. 1	800	2447	incluant 1916 instructions du noyau système i.e. 78 %
	hyp. 2	1000		
reception	hyp. 1	1250	1563+N	incluant 971 instructions du noyau système i.e. 57%
	hyp. 2	1500		

\* N est pris égal à 128 dans le cas de l'implantation A.

Une étude similaire a été effectuée à IBM Zurich [IBM 86]; les résultats de l'évaluation de la complexité d'un algorithme implantant l'émission ou la réception d'une trame d'une couche de Transport sont du même ordre de grandeur que ceux obtenus dans notre étude.

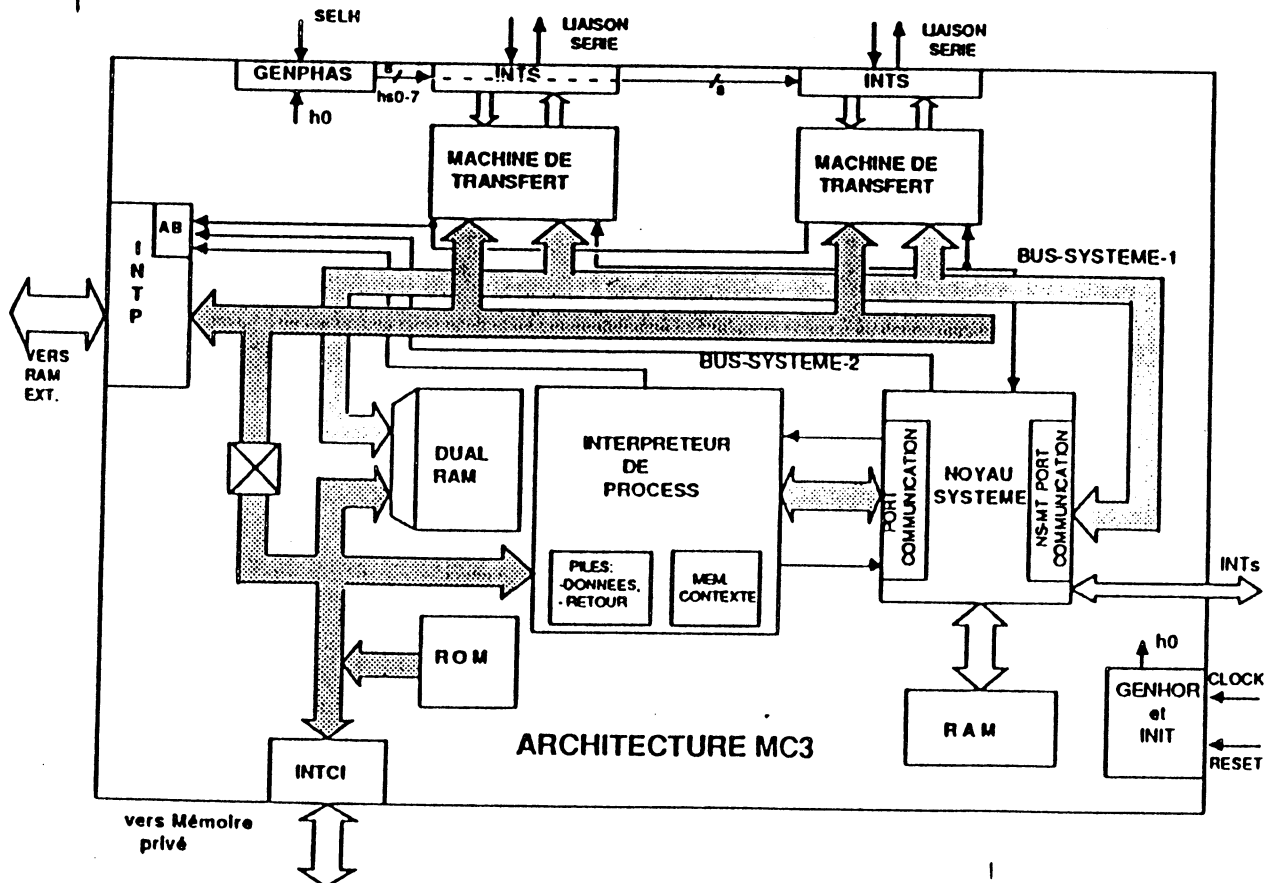


### 3.22. Architecture fonctionnelle cible

Le modèle d'architecture fonctionnelle suivant a été adopté, comme conséquence directe de la synthèse de l'analyse menée dans le chapitre précédent. Nous faisons appel à six processeurs qui sont dédiés chacun à une tâche différente :

+ Un processeur câblé, appelé NS pour *Noyau Système*, gère la commutation des processus et la communication des processus et des processeurs, la gestion des horloges de garde et le traitement des interruptions. La synchronisation entre processus est conçue comme une forme dégradée de communication, selon la définition donnée dans CSP [HOA 79]. Il n'y a pas de différence faite par NS entre une entité processus logique et une entité processeur physique dans la gestion de la communication entre deux entités processus.

+ Un microprocesseur à architecture RISC, appelé IP pour *Interpréteur de Processus*, qui exécute le code des processus (i.e. les programmes qui implantent les protocoles). Les dits programmes se trouvent *en partie* figés dans une mémoire interne ROM du circuit dont le bus d'accès est propre à l'IP. Ce bus privé peut être prolongé vers l'univers extérieur au circuit soit pour augmenter les capacités de mémoire programme, soit pour des raisons propres à l'architecte système qui conçoit des coupleurs à partir du circuit MC3 (l'installation des programmes et des données en mémoire RAM est une nécessité système pour beaucoup de bureaux d'études industrielles).



Les principales caractéristiques de l'IP sont les suivantes :

- . son jeu d'instructions réduit est adapté à l'implantation des protocoles de communication;
- . les primitives système du jeu d'instructions de l'IP sont exécutées par le Noyau Système et
- . la commutation de contexte est gérée par un mécanisme câblé.

+ Ces deux processeurs, IP et NS, se déchargent en grande partie de leur tâche de gestion d'interfaces sur des machines intelligentes qui gèrent le passage d'information d'un niveau de protocole  $n$  à un autre niveau de protocole  $n+1$  (ou  $n-1$ ). Ces *machines intelligentes de transfert*, appelées MT, sont dédiées uniquement à la gestion d'interface.

Supposons que la couche la plus basse gérée dans MC3 soit la couche  $b$  et la couche la plus haute  $h$ , MC3 doit gérer les interfaces  $(b, b-1)$  et  $(h, h+1)$ . Chaque interface nécessite un canal d'émission et un canal de réception et chaque canal est géré par une MT de type réception ou de type émission. Chaque canal peut être :

- . soit une ressource partagée avec d'autres processeurs internes de MC3 et avec le microprocesseur externe : c'est le cas du *bus mémoire du microprocesseur*, si le transfert doit avoir lieu entre deux zones d'un même espace mémoire locale ou entre deux espaces mémoire séparés par un bus fond-de-panier;

- . soit une ressource dédiée telle qu'une ligne série, si le transfert doit avoir lieu entre un espace mémoire local et un microprocesseur possédant une interface série (c'est le cas du microprocesseur INMOS TRANSPUTER).

### 3.23. Architectures des systèmes utilisant MC3

Actuellement, les architectures des systèmes implantant les couches hautes de l'OSI sont bâties en tenant compte de l'absence de composant circuit intégré gérant ces couches. Les exemples d'implantation matérielle sous forme de cartes montrent cependant l'approche suivie par les architectes [OST 87] [APT 87.1] [THO 87].

Dans quel environnement système un composant tel que MC3 peut-il être utilisé ?

Répondre à cette question nous conduit à faire :

+ des conjectures sur la manière dont les architectes système vont utiliser MC3 ; elles sont cependant basées sur les architectures des cartes existantes où l'on intégrerait les fonctions de plusieurs composants dans un seul ; et

+ des projections dans l'avenir sur les besoins futurs des architectes système sachant qu'ils vont avoir à leur disposition des composants microprocesseurs plus puissants que les processeurs actuels, ne serait-ce qu'en terme de gestion du parallélisme, en terme de nombre d'instructions (et d'opérations) exécutées par seconde ou en terme de nouvelles interfaces physiques.

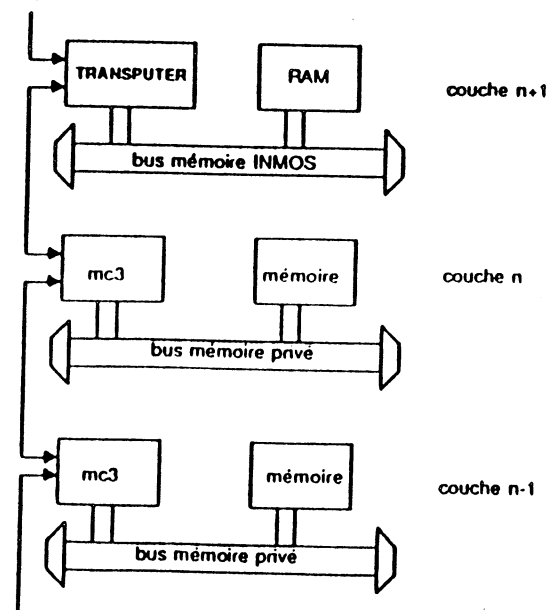
Il existe a priori deux types de solutions.

#### 3.231. ARCHITECTURE MASSIVEMENT PARALLELE

L'environnement externe, qui peut dialoguer avec MC3 via une ligne série, est supposé être celui du TRANSPUTER d'INMOS ou bien un autre composant MC3. Une architecture générique possible qui utilise des composants MC3 est montrée dans la figure ci-contre. Dans cette figure, le scénario suivant est suggéré :

. si la valeur de n est par exemple 5, cela signifie que la couche Application et éventuellement la couche Présentation tournent sur le microprocesseur TRANSPUTER et

. les couches Session et Transport tournent chacune sur un composant MC3.



Architecture massivement parallèle :  
chaque couche est gérée par un composant MC3

Ce type d'architecture offre à chaque couche de puissantes possibilités de calcul car la séparation du niveau de traitement augmente les ressources propres de chaque composant. Le nombre de connexions à traiter par MC3 peut être plus élevé. Le temps de traitement à l'intérieur de chaque couche est plus rapide dans la mesure où il y a moins de processus à ordonnancer.

Le parallélisme supplémentaire de type pipe-line introduit permet de concevoir des architectures modulaires, avec une cohabitation à la carte entre différents protocoles de niveaux différents.

Cependant, ces architectures impliquent qu'il n'y a pas de partage de la principale ressource : la mémoire. Le transfert des trames d'une couche à l'autre doit être nécessairement réalisé par recopie de l'information à chaque traversée de l'interface des deux couches. Recopie qui est en fait très handicapante, nonobstant le débit élevé d'une liaison série du type INMOS (débit nominal de 14,54 M bits/s).

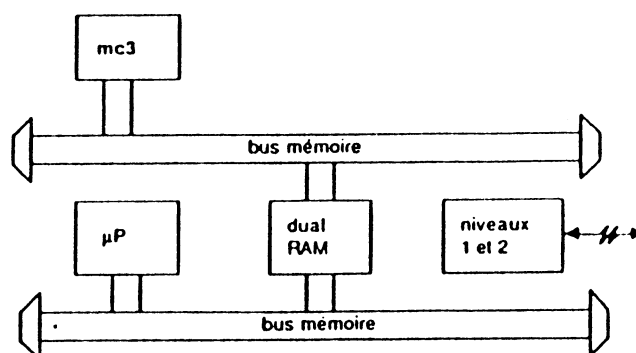
### 3.232. ARCHITECTURES PSEUDO-PARALLELES

L'environnement externe, qui peut dialoguer avec MC3 via un bus parallèle, est supposé être celui du microprocesseur 68020 de MOTOROLA qui est le microprocesseur cible défini par le cahier des charges du composant MC3. Le principe de la communication entre le microprocesseur et MC3 est celui d'une communication asynchrone par partage de ressource mémoire. Dans l'architecture modulaire du composant MC3, l'adaptation à un microprocesseur donné est réalisée uniquement et spécifiquement par le module *Interface Parallèle*, appelé INTP. Dans le cas où il s'agit de concevoir un nouveau circuit MC3 compatible avec un autre microprocesseur, par exemple le microprocesseur 80386 d'INTEL, l'effort concernera principalement la conception d'un nouveau module INTP.

#### A/ Utilisation du bus parallèle du microprocesseur

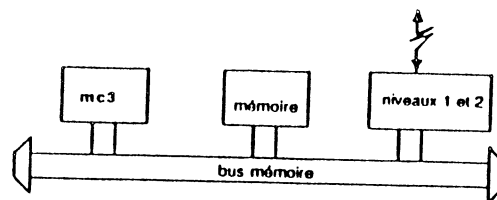
Les architectures possibles sont montrées dans les deux figures ci-contre. Le scénario suivant est suggéré, parmi d'autres scénarios possibles :

. les niveaux bas (2 et 1) sont gérés soit par le couple conventionnel d'un microprocesseur et d'un circuit de communication (ou un coupleur de connexion) de niveau MAC, soit par un circuit de communication de niveau LLC avec sa mémoire partageable; les structures de données en réception (respectivement à l'émission) doivent être converties du format du circuit de niveau bas vers un format compatible MC3 par le microprocesseur (respectivement dans l'autre sens).



MC3 gère plusieurs couches de protocole de communication

. la couche Réseau (ou/et les couches plus hautes) tourne sur un composant MC3 qui partage de la mémoire via le même bus microprocesseur ou qui partage de la mémoire à double accès; cette architecture sous-entend que MC3 doit avoir des programmes pour gérer directement les circuits de niveaux 1 et 2.



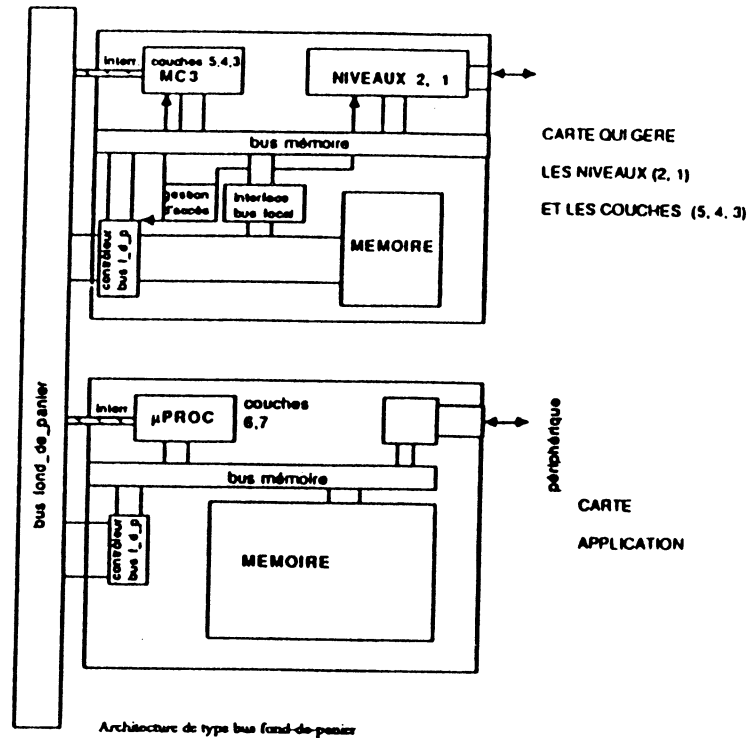
Mentionnons que les programmes qui sont exécutés par MC3 sont situés soit dans une mémoire ROM interne à MC3 soit dans des mémoires RAM et ROM externes accessibles par un bus privé différent du bus microprocesseur. Par conséquent, les conflits d'accès au bus microprocesseur proviennent de son utilisation pour le trafic des données et des instructions du microprocesseur, et dans le pire cas, de son utilisation conjointe pour le trafic des données de MC3.

Il est donc nécessaire de minimiser les accès à la mémoire partageable pour manipuler les trames. Cette contrainte nous a amené à définir une structure de donnée spécifique, appelée PnPDU ("Processing level n Protocol Data Unit"), qui permet à n'importe quel processus de MC3 d'accéder à ses propres trames sans avoir à copier ou à traduire des zones d'informations. Nous avons choisi d'implanter cette structure de données dans une mémoire RAM interne au composant MC3 (§ 3.24).

Il est aussi nécessaire de minimiser les conflits d'accès au bus microprocesseur. Comme conséquence de cette contrainte, les DMA intégrés des Machines de Transfert fonctionnent systématiquement en mode "burst", en corrélation avec la taille des FIFO. Ce mode de fonctionnement qui s'exprime avec seuls paramètres (@\_début de la zone à écrire ou lire, @\_fin de ladite zone) est géré par les DMA et l'INTP. Rappelons que chaque Machine de Transfert est armée par un processus de l'IP; après quoi, ses DMA doivent accéder aux différentes unités de données de service ou unités de données de protocole pour transférer de manière autonome les trames.

## B/ Utilisation d'un bus fond-de-panier

Une des caractéristiques de l'utilisation d'un bus fond-de-panier est le fait qu'il y a deux espaces mémoire séparés. Si le microprocesseur se trouve sur une carte et le composant MC3 sur une autre carte, il faut qu'ils puissent se communiquer entre eux; i.e. que le circuit MC3 agisse comme un maître sur son bus local et puisse offrir des lignes d'interruption pour la synchronisation avec le microprocesseur. Une architecture possible de type fond-de-panier qui utilise un composant MC3 est montrée dans la figure ci-contre. La gestion de la communication avec le microprocesseur via le bus fond-de-panier est effectuée par un processus de l'IP.



Le transfert des trames de la couche 5 vers la couche adjacente est nécessairement réalisé par recopie d'informations à travers le bus fond-de-panier. Il y a beaucoup de similitudes entre cette architecture et celle présentée dans le paragraphe § 3.231, si l'on remplace le bus fond-de-panier par une ligne série et que le microprocesseur est un TRANSPUTER.

### 3.233. CONCLUSION

Les environnements système dans lesquels le composant MC3 peut être utilisé ne sont pas limités aux architectures système présentés ci-dessus. Un autre exemple possible consiste à utiliser MC3 comme gestionnaire de passerelle entre un réseau local et un réseau général. Dans cette configuration, on confie à MC3 le soin de gérer la couche X25.3 avec un nombre important de voies logiques sachant que les niveaux 2 et 1 sont gérés par des composants spécifiques.

En conclusion, l'objectif recherché dans la proposition des différents types d'architectures est d'abord de servir de base de travail pour les architectes qui implanteront les applications de communication qui utilisent le composant MC3, et d'autre part de nous servir comme modèles pour la mise au point des fonctions de communication du composant MC3, communication interne à MC3 ou communication entre MC3 et l'univers extérieur.

### 3.24. Complexité des structures de données

Il s'agit de définir les structures de données spécifiques manipulées par le composant MC3 pour la gestion des trames entrantes (sous forme de nSDU), des trames gérées par les entités communiquant des couches (sous forme de nPDU) et la gestion des trames sortantes (sous forme de nSDU). Cette définition doit tenir compte du fait que le composant MC3 peut être destiné à implanter *une ou deux ou trois couches* préalablement définies.

Selon les hypothèses du cahier des charges, toutes les fonctionnalités recommandées par les comités internationaux de standardisation pour les trois couches citées doivent être prises en considération par le composant MC3. En particulier, *les structures de données définies dans MC3 doivent permettre l'implantation de la segmentation d'une trame, de la concaténation des différentes trames et de leur multiplexage à l'émission à travers les différentes couches implantées (respectivement le réassemblage, la séparation et le démultiplexage à la réception)*. Il faut rappeler que ces fonctionnalités sont propres à chaque couche et que leur sémantique diffère d'une couche à l'autre.

Selon les objectifs que nous nous sommes fixés dans le projet et que nous avons eu l'occasion de justifier par ailleurs, les structures de données définies dans MC3 doivent d'autre part répondre aux assertions suivantes.

- + Elles doivent être *indépendantes* de la couche qui s'exécute et leur définition *récurrente* si deux ou plus de deux couches adjacentes doivent être exécutées ensemble dans MC3.

- + La définition des algorithmes qui gère ces structures de données doit *minimiser les fonctions de recopie de données* d'une zone mémoire vers une autre zone mémoire, sachant que la majeure partie du temps d'exécution d'une couche de protocole donnée est passée dans l'exécution de quelques fonctions dont la fonction de recopie de données (ou de mouvement de blocs de données) [KAN 86]. Par conséquent, les algorithmes de manipulation des trames à l'intérieur du composant MC3 (séparation des en-têtes, assemblage, désassemblage, ...) se feront sans copie des données.

Chaque hypothèse prise individuellement ne constitue pas une nouveauté et a déjà été prise en compte maintes fois par différentes implantations. Il en est de même pour des sous-ensembles de ces hypothèses; en voici, par exemple, un cas caractéristique : minimisation des recopies de données, séparation et démultiplexage en réception, absence de segmentation, de concaténation et de multiplexage à l'émission.

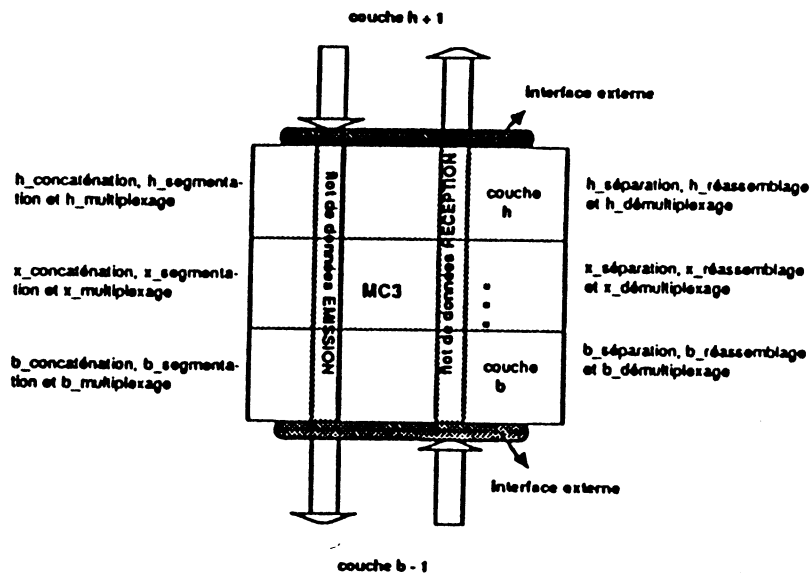
L'apport de ce travail réside dans le fait que, dans l'étude des structures de données de MC3 [DAN 88.1], l'ensemble de ces hypothèses est pris en compte globalement.

### 3.241. LES FLOTS DE DONNEES EN EMISSION ET EN RECEPTION

Soit  $b$  l'appellation de la couche la plus basse implantée dans MC3 et  $h$  l'appellation de la couche la plus haute. L'analyse d'un circuit de communication de donnée tel que MC3 montre qu'il y a deux principaux flots de données : une en Emission et une en Réception.

. Le flot Réception provient d'une trame issue de la couche  $b-1$ , qui sera reçue par l'interface  $(b-1, b)$  de MC3; après le traitement interne de MC3 et le franchissement logique des couches intermédiaires entre  $b$  et  $h$ , ce flot de données se termine par l'émission d'une trame vers la couche  $h+1$ .

. Le flot Emission suit le chemin inverse.



Dans cet exposé, nous allons décrire principalement le flot de données Réception, car les trois fonctionnalités mentionnées ci-dessus sont plus souvent employées en réception. Une fonctionnalité telle que la concaténation de la couche Transport est seulement optionnelle selon les recommandations du standard CNMA [ESP 86].

### 3.242. GESTION DE LA MEMOIRE A L'INTERFACE EXTERNE

La communication entre MC3 et l'univers extérieur s'effectue par partage de ressources communes; ressources qui peuvent être soit une ligne 'série', soit un bus microprocesseur parallèle, ainsi que des lignes d'interruption [MC3 87.2]. Rappelons que MC3 se présente vis-à-vis du microprocesseur comme un coprocesseur et non comme un esclave sur le bus parallèle.

Les trames de service que le composant MC3 reçoit de l'univers extérieur à partir de la ligne série (ou partage avec cet univers extérieur via le bus microprocesseur) sont organisées à l'interface  $(b-1, b)$  sous une forme de liste classique : chaque trame est décrite par un descripteur qui contient le pointeur d'accès à une liste de descripteurs de tampons qui décrivent les tampons de données. A



quelques différences près (liste sans fin, liste circulaire, tampon de taille variable ou fixe, ...), les principaux constructeurs de circuits de communication tels que INTEL [INT 84.1], MOTOROLA [MOT 86], AMD [AMD 85], WD [WD 84] ont choisi cette forme de structure de données pour l'échange des données à l'interface.

A cause des objectifs généraux de MC3, il n'y a pas d'hypothèses faites sur la correspondance entre frontières physiques d'une liste de tampons avec frontières logiques d'une trame : (en-tête, champ de paramètres, champ de données,...). Cependant, la définition des structures de données de MC3 permet la prise en compte de telles hypothèses en fonction de l'utilisation du composant MC3 : facilités d'organisation de la mémoire à partir du système d'exploitation du microprocesseur de la couche adjacente, support de synchronisation pour la communication asynchrone entre le système d'exploitation du microprocesseur et MC3, gestion d'une seule couche, etc...

Les trames de service que le composant MC3 envoie vers l'univers extérieur sont bâties à partir des structures de données décrites par les PnPDU; ces structures de données sont homogènes avec celles prévues pour la gestion des interfaces logiques internes à MC3, présentées ci-après. Un processus tournant dans l'IP bâtit une trame de service et commande son envoi à la machine de transfert responsable de cette interface.

### 3.243. GESTION DE LA MEMOIRE AUX INTERFACES LOGIQUES INTERNES A MC3 LA STRUCTURE DES PnPDU ET LES ALGORITHMES ASSOCIES

Le traitement du flot de données en Réception peut introduire : (i) la nécessité de démultiplexer différentes nPDU, (ii) de les séparer et (iii) d'assembler plusieurs segments d'une nSDU. Comme les fonctionnalités de segmentation, de concaténation et de multiplexage du flot de données d'Emission sont potentiellement utilisables à travers les différentes couches du site d'émission, le squelette de la trame reçue ne peut pas être connu à l'avance. L'exemple présenté ci-dessous (§3.243.B) de quelques squelettes de trames est caractéristique du phénomène.

Pour manipuler les différentes trames et les différents champs des trames à l'intérieur de MC3, sans avoir recours aux fonctions de recopie, nous avons défini une nouvelle structure de données, appelée PnPDU ("Processing level n Protocol Data Unit").

Toute trame d'une couche n donnée qui est traitée par le composant MC3 est définie à travers un PnPDU.

DEFINITION DU PnPDU COURANT

type_PDU	réserve besoin futur
ét_succes_logique	@_succes_logique
type_succes_physique	@_succes_physique
lg_nPDU	@_debut_nPDU
Int_nPDU	@_Pn-1_PDU
lg_en_tete_nPDU	@_fin_nPDU

**A/ Définition du type PnPDU**

La définition complète du type PnPDU a été effectuée en plusieurs étapes qui correspondent aux différents niveaux d'abstraction suivants [LAR 87] (ces niveaux sont vus du côté de la conception des circuits) :

(i) niveau architectural et algorithmique : spécifications de la structure de données et des algorithmes associés de manipulation [MC3 87.1];

(ii) niveau fonctionnel :

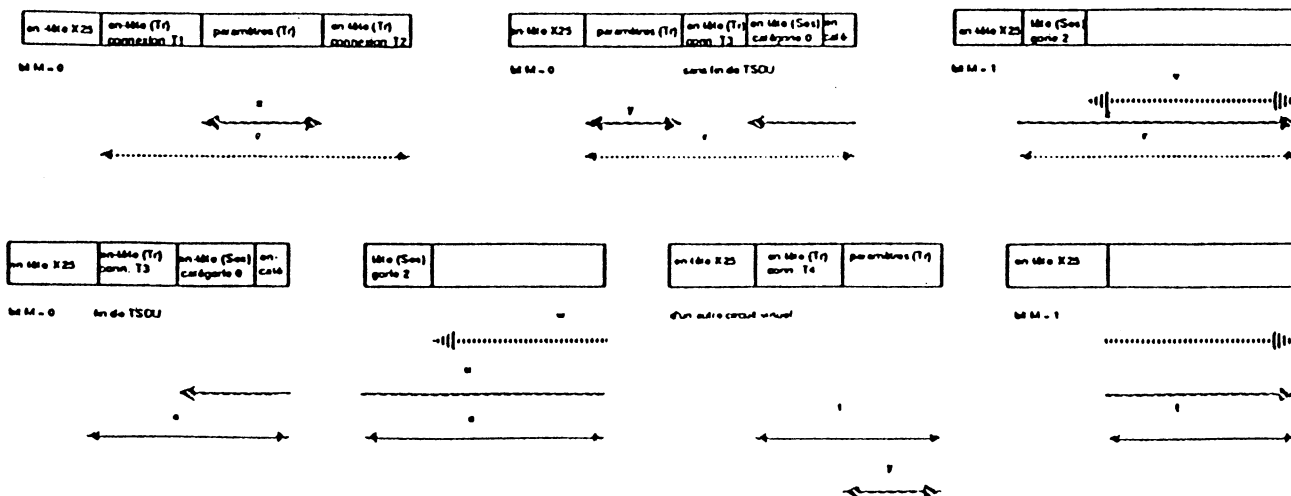
. description comportementale : sur une maquette fonctionnelle (à base de composants TRANSPUTER, et avec le langage de programmation parallèle OCCAM) qui simule le comportement du composant MC3, la simulation fonctionnelle du type PnPDU a été effectuée dans le cadre d'une mini-application d'étude de performances (§ 4.22);

. description structurelle : l'architecture matérielle du DMA qui gère les émissions de trames décrites par des PnPDU a été définie en terme de registres, d'UAL, de PLAs, de séquenceur, etc . );

(iii) niveau logique : cette architecture matérielle a été ensuite définie en terme de portes de base ou portes complexes et sa simulation logique effectuée (§ 4.22).

**B/ Utilisation de PnPDU dans un scénario de réception**

Nous allons montrer grâce à un exemple simplifié comment opèrent les procédures qui définissent le type PnPDU. Supposons que la couche basse gérée par le composant MC3 est la couche X25 niveau Paquet (b=3) et la couche haute la couche Session (h=5). Supposons que les trames qui arrivent du niveau 2 soient les suivantes : il y a successivement 7 trames X25 de deux circuits virtuels différents; ces trames contiennent 5 trames de la couche Transport de 4 connexions différentes; ces trames de la couche Transport contiennent deux trames de la couche Session d'une seule connexion Session. Ces trames possèdent les squelettes suivants :



Le processus de réception de la couche X25 bâtit les P<sub>3</sub>PDU correspondant aux trames X25, trames qui ont été extraites des SDU du niveau 2 (les en-têtes service du niveau 2 ne figurent pas dans l'exemple montré ci-dessus, pour rendre le schéma plus clair). Ce processus a tenu compte des trames qui sont spécifiques à la couche X25 (ne contenant pas des données des couches supérieures) et les a déjà traitées.

A partir des NPDU décrites par les P<sub>3</sub>PDU, l'interface interne à MC3 entre les couches X25 et Transport bâtit des NSDU, à partir desquels le processus de réception de la couche Transport bâtit les P<sub>4</sub>PDU correspondant aux TPDUs. Ce scénario est répété entre la couche Transport et la couche Session. Précisons que chaque SPDU de catégorie 2 est concaténé dans cet exemple à un SPDU de catégorie 0.

La figure de la page suivante montre le résultat complet des trois couches de P<sub>n</sub>PDU. La couche Session finit par préparer un SSDU pour l'interface externe (5,6) et livre à la machine de Transfert Emission (§ 3.242) l'adresse du P<sub>5</sub>PDU correspondant comme unique paramètre.

Note : Les P<sub>n</sub>PDU sont destinés principalement à décrire des nPDU. Comme le mécanisme d'analyse des nPDU ainsi que le mécanisme de constitution des nPDU (à l'émission, qui n'est pas décrit ici) peuvent être appliqués de manière identique aux nSDU, les P<sub>n</sub>PDU sont utilisés temporairement pour décrire des nSDU.

### **C/ Résultats**

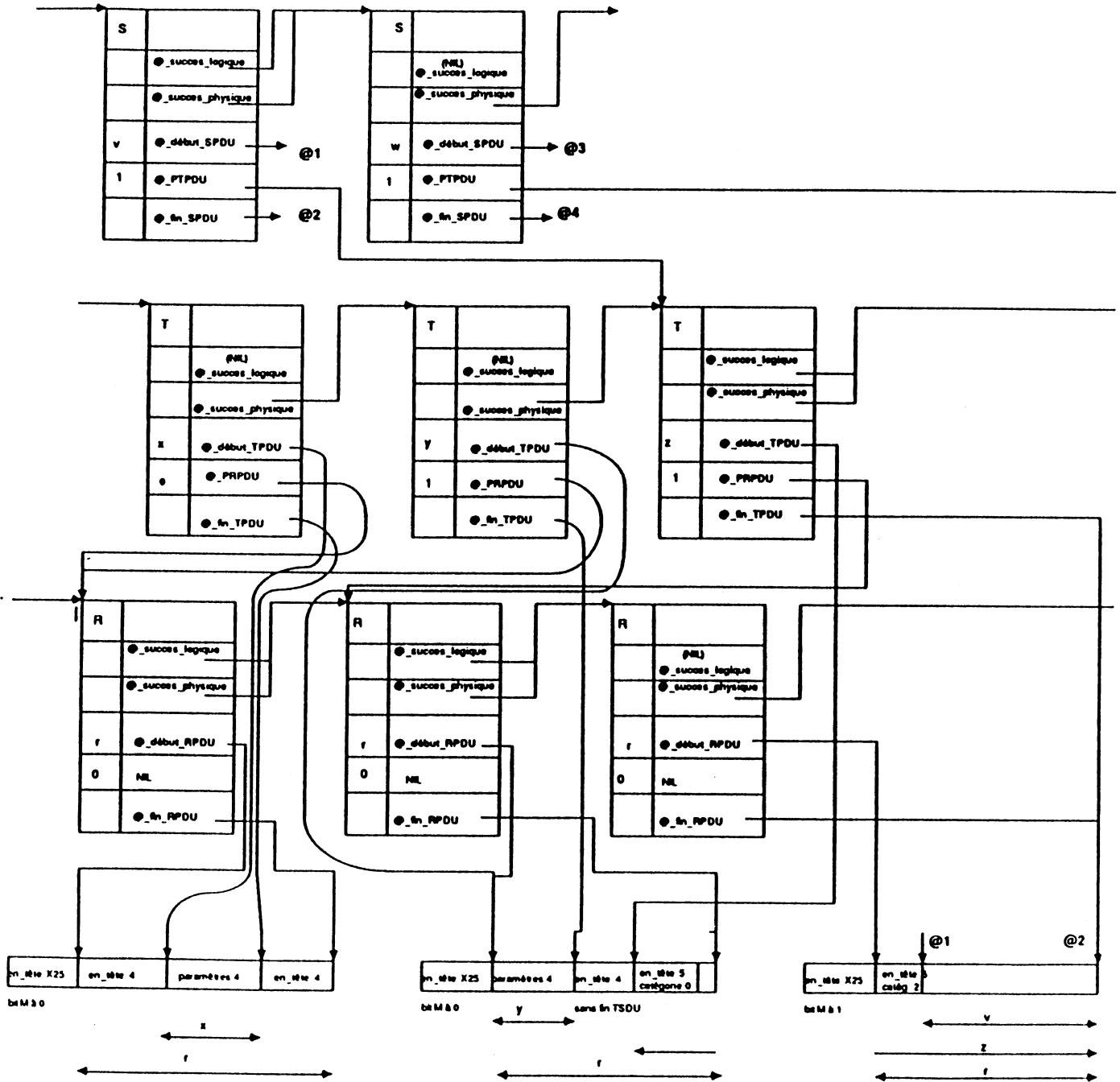
Une analyse et des simulations d'une complexité équivalente ont été menées pour le flot de données Emission, en utilisant la même définition des P<sub>n</sub>PDU.

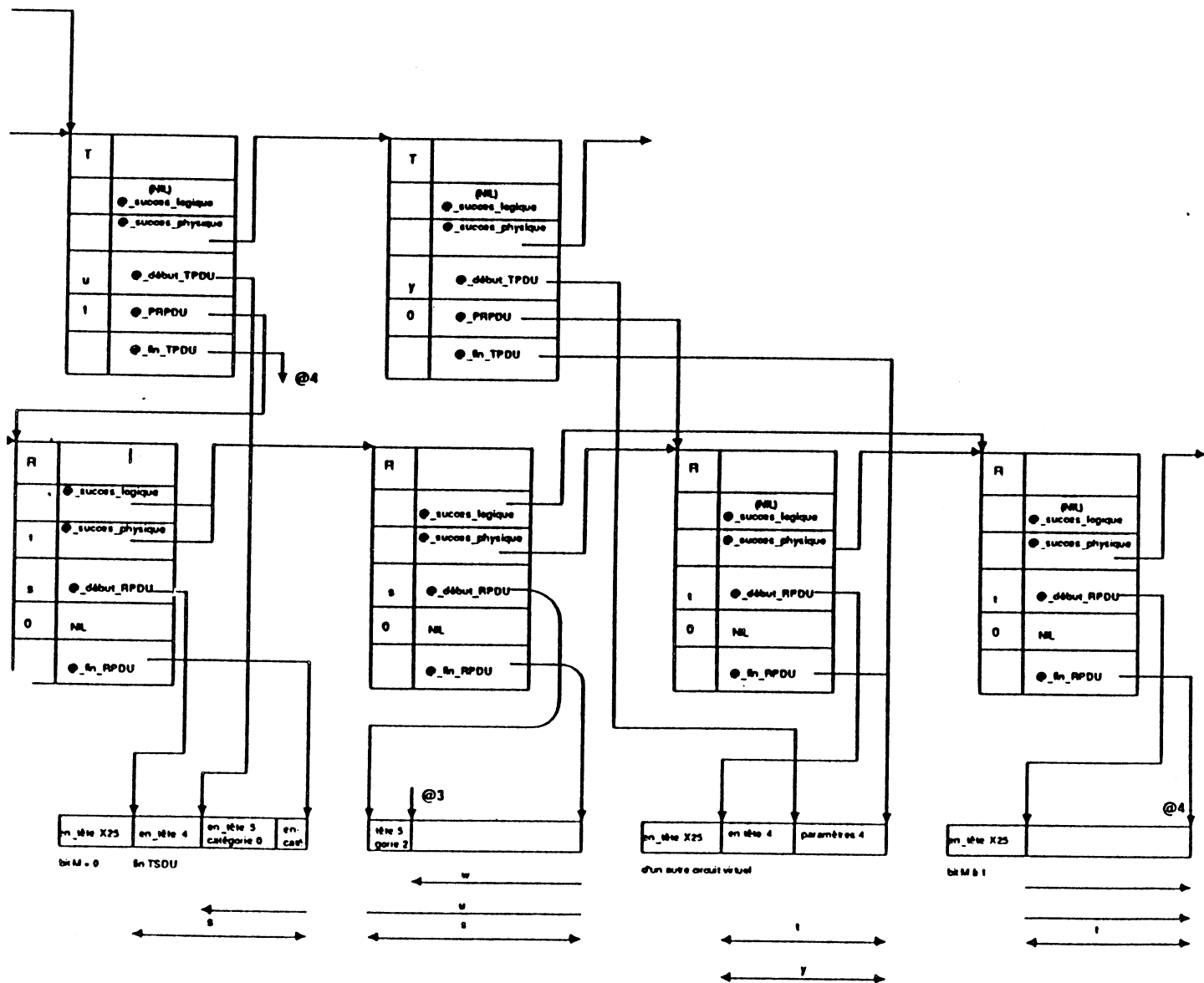
Les résultats obtenus lors des simulations comportementale, fonctionnelle (§ 4.221) et logique (§ 4.222) vérifient les assertions initiales.

**R1.** La définition des P<sub>n</sub>PDU est indépendante de la couche traitée.

**R2.** Les algorithmes de parcours (analyse en réception et constitution en émission) de l'arbre des P<sub>n</sub>PDU sont récursifs.

**R3.** Avec la structure de donnée des P<sub>n</sub>PDU, tout processus de toute couche implantée par MC3 peut accéder à ses propres trames nPDU sans avoir à copier ni à translater quelque zone de mémoire que ce soit.





### 3.25. Architecture des Machines de Transfert

Rappelons quelques principales propositions retenues pour la définition de l'architecture fonctionnelle de MC3 :

- . parallélisme entre le traitement de l'interface et le traitement des éléments du protocole,
- . implantation optimisée de la gestion mémoire (en particulier les fonctions de manipulation de la mémoire dans le traitement interne des PDU d'une couche ou dans le traitement interne des PDU et un sous-ensemble de SDU de plusieurs couches adjacentes).

Les machines de transfert ainsi que les structures de données qu'elles partagent avec les processus de l'IP constituent un ensemble de réponses à ces propositions.

#### 3.251. PARALLELISME ENTRE LES MACHINES DE TRANSFERT ET LES PROCESSEURS INTERPRETEUR IP ET NOYAU NS

Le principe de la communication entre le processeur IP et une machine de transfert MT est basé sur une communication de message en mode maître à esclave; cette communication est implantée sur des canaux spécifiques. Le scénario suivant de l'émission d'une trame vers la couche h+1 ou b-1 montre le fonctionnement d'une telle communication :

- . Un processus autorisé, qui est interprété par l'IP, arme une MT d'émission donnée en lui envoyant une commande avec pour paramètre l'adresse du PnPDU qui définit la trame que la MT aura à émettre. L'arbre des descripteurs de PnPDU définissant la trame aura été au préalable préparé par le dit processus.

- . La MT exécute la commande et envoie un compte rendu au processus de l'IP.

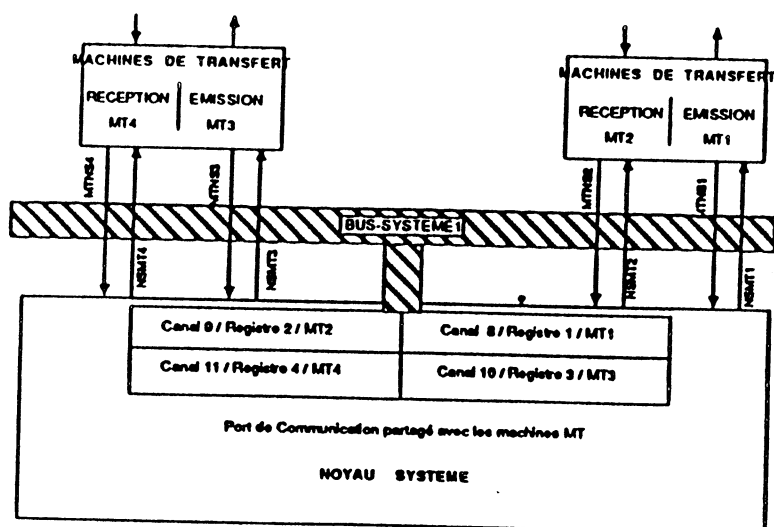
Dans le cas de la réception d'une trame, la MT réception signale à un processus dédié (via un canal donné) les événements tels que réception de message à partir d'une ligne série, détection d'erreur de transmission, ...

Chaque MT est associée à un canal prédéfini. Ces canaux, comme tout autre canal, sont gérés par le noyau système NS qui les considère :

- . comme des canaux banalisés pour la communication des commandes et des comptes rendus entre l'IP et NS, et

- . comme des registres spécifiques qui forment un port de communication pour l'échange de messages avec les MT via le bus système interne 1; ces registres sont adressables par les MT en lecture (des commandes) et en écriture (des comptes rendus).

La figure suivante montre les canaux et les signaux utilisés pour la communication entre l'IP et les MT.



L'IP et les MT partagent l'accès à la mémoire des données et des descripteurs :

- . Les trames nSDU ou nPDU sont localisés dans la mémoire externe. Le chemin d'accès est constitué par le bus système 2 et l'INTP.

- . Les descripteurs PnPDU sont localisés dans une Dual RAM interne au composant MC3, pour des raisons d'optimisation du temps d'accès. Le chemin d'accès est constitué par le bus système 1 et un arbitre de bus commun entre l'IP et les MT.

### 3.252. ARCHITECTURE DES MACHINES DE TRANSFERT

Il y a deux types de machines de transfert : les MT de réception et les MT d'émission.

Les MT de réception reçoivent la trame via la ligne série et la stockent dans la mémoire externe via le bus parallèle du microprocesseur sous une structure de liste préalablement préparée.

Les MT d'émission ont deux modes de fonctionnement différents : selon la commande qu'elles reçoivent, elles se configurent pour une émission vers le médium 'série' ou pour une émission vers la mémoire externe via le bus parallèle du microprocesseur.

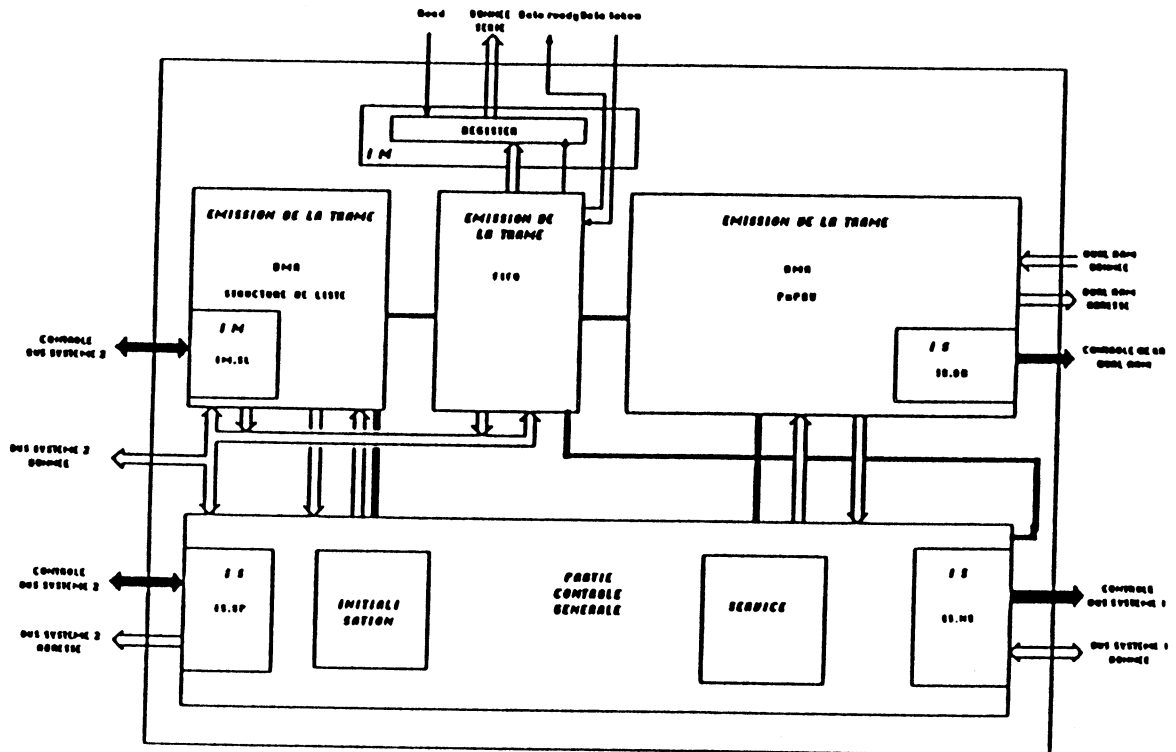
#### A/ Modèle d'architecture

L'architecture fonctionnelle d'une MT est définie selon le modèle exposé pour les circuits de communication implantant les niveaux 1 et 2.MAC. Il s'agit d'un ensemble d'unités asynchrones ou de processus communiquant dont les caractéristiques ont été définies dans les sous-chapitres 2.32 et

2.34.

## B/ Architecture de la MT d'émission

L'architecture de la MT la plus complexe qui est la *MT d'émission* se compose de cinq processus :



Architecture de la machine de transfert Émission

. Le processus *Interface avec le Medium IM* est composé de deux autres processus qui sont exclusifs :

+ S'il s'agit d'une émission série, l'interface avec le medium 'série' est gérée le processus d'interface avec INTS.

+ S'il s'agit d'une émission parallèle, l'interface avec le medium parallèle est gérée par le processus IM.SL (Gestion du Bus Système 2 pour l'émission vers une Structure de Liste).

. Le processus *Interface avec la Station IS* est composé de plusieurs processus :

+ Le processus IS.NS (Gestion du Bus Système 1 pour la communication avec NS) gère la communication des messages de commande et de compte-rendu entre la MT et NS.

+ Le processus IS.DR (Gestion du Bus Système 1 pour la communication avec la Dual RAM) gère la lecture des descripteurs PnPDU localisés dans la Dual RAM.

+ Le processus IS.SP gère le Bus Système 2 pour prendre dans la mémoire externe les trames décrites par la Structure des PnPDU en vue de l'émission vers le medium 'série' ou parallèle.



Notes : Les processus IS.SP et IM.SL partagent le bus système 2. Le conflit d'accès à ce bus est géré par un processus arbitre de bus propre à la MT d'émission.

Les processus IS.NS et IS.DR sont exclusifs dans le temps.

. Le processus *Gestion de la Trame* est en fait un processus d'Emission de la Trame. Il se compose du sous-ensemble de processus suivants :

+ Traitement des commandes de l'IP (via NS) en vue de l'émission d'une trame. Ce processus échange des informations de contrôle avec le processus IS.NS.

+ Transfert des informations :

\* Processus Accès à la Mémoire des trames décrites par les PnPDU (appelé dans la figure DMA PnPDU). Ce processus échange des informations de contrôle avec les processus IS.DR et IS.SP.

\* Processus Tampon (appelé dans la figure FIFO). Ce processus échange des données soit avec INTS, soit avec le processus Accès à la Mémoire de destination.

\* Processus Accès à la Mémoire de destination (appelé dans la figure DMA Structure de Liste) qui est organisée en structure de liste. Ce processus n'est actif que dans le cas d'une émission parallèle.

+ Communication avec le processus Interface avec le Medium.

+ Gestion des Erreurs.

. Le processus *Initialisation* .

. Le processus *Service* .

### C/ Observations

La définition fonctionnelle puis logique de l'architecture d'une machine est suivie, de manière générale, par la validation de ladite architecture. Effectivement dans notre cas, cette validation est effectuée par simulations (§ 4.22).

Dans le cycle souvent recommencé de la phase (définition, validation, évaluation), l'évaluation des performances de la machine de transfert a été effectuée (§ 4.22)1 [DIO 88]. Signalons cependant que le débit de la machine de transfert est pour l'instant borné par le débit de la ligne série INMOS, aussi bien en émission qu'en réception.

### 3.253. SPECIFICITE DU CONCEPT MACHINE DE TRANSFERT

La Machine de Transfert remplit le rôle d'un *DMA intelligent* qui gère la communication entre un microprocesseur et un circuit de communication (en l'occurrence MC3) ou entre un microprocesseur et un coupleur de connexion (où *la MT constituera un composant spécifique*).

La spécificité d'un tel composant est qu'il est non seulement capable de traiter des structures de liste de communication, mais qu'il est aussi et surtout *capable d'assembler dynamiquement des éléments constituant d'une trame, qu'elle soit une PDU ou SDU, et quelle que soit la structuration de son squelette*.

### 3.3. VERS UN MICROPROCESSEUR DE COMMUNICATION

L'approche que nous préconisons pour l'implantation matérielle des protocoles de haut niveau est, comme nous l'avons vu, une approche de type *microprocesseur à architecture parallèle dédié à un ensemble d'applications données*.

Le composant MC3 est dédié à l'implantation des protocoles des couches Réseau, Transport et Session, qu'elles soient celles des recommandations CCITT ou celles des standards ISO ou celles d'autres standards (MAP, ECMA, DOD, ...).

+ Les architectures fonctionnelle et logique de MC3 sont dédiées au domaine cité, car la démarche adoptée pour les définir était une démarche déductive.

. Il faut retenir que l'IP est un microprocesseur à architecture RISC dont les instructions ont été choisies pour être adaptées à l'implantation des protocoles des couches hautes et pour que l'exécution des programmes écrits dans ce langage soit performante.

. NS est une machine câblée qui fonctionne en parallèle avec l'IP. NS a pour tâche principale de gérer les processus exécutés par l'IP, leurs communications, leurs synchronisations et les chiens de garde.

L'étude de l'Interpréteur de Processus IP et du Noyau Système NS a été effectuée par le département d'Etudes de la société APTOR. De ce fait, cette étude [APT 87.4] n'est pas décrite dans ce rapport. Notre équipe a participé en expertise dans les choix des principales instructions de l'IP et des principales fonctionnalités de NS.

+ L'évaluation de la taille des mémoires internes ou privées du composant MC3 tient compte des contraintes suivantes du cahier des charges :

. Le composant MC3 doit pouvoir implanter les trois couches de protocoles 3, 4, 5 ISO ou équivalentes. Nous estimons que l'ensemble des programmes des trois couches écrits en langage MC3 doit être logeable dans un espace mémoire adressable par un registre de 16 bits (64 K octets). Les programmes figés en ROM interne du composant MC3 occupent 16 K octets.

. Le nombre maximum de connexions logiques gérées par MC3 pour une couche donnée est fonction du nombre de couches implantées; la mémoire n'est pas partitionnée en différentes zones réservées aux différentes couches.

. La taille de la mémoire Dual RAM (réservée au PnPDU) est calculée en fonction du nombre maximum de connexions logiques gérées par MC3 pour une unique couche X.25 paquets. Elle est en l'occurrence de 2,5 K octets.

. Nous avons décidé, au vu de la complexité du composant et dans l'état actuel de la technologie, de borner la décomposition des applications en 64 processus au maximum, de borner le nombre maximum de canaux utilisables par les processeurs et les processus (256) et de borner le

nombre maximum de chiens de garde armés (256).

+ La définition architecturale du composant MC3 a été effectuée à différents niveaux : (i) niveau fonctionnel et niveau logique. La validation de ces différents niveaux a été menée à bien par simulation fonctionnelle (comportementale) [MC3 88.5] et simulation logique [MC3 88.3]. L'évaluation des performances a fait l'objet d'un travail spécifique [MC3 88.6] [DIO 88]. Ces différents aspects sont traités dans le chapitre 4.22, y compris l'évaluation des performances, pour des raisons de facilité de présentation.

Parallèlement avec la prise en compte au niveau architectural des enseignements des études de validation et de performances, l'aspect du génie logiciel dans le développement de programmes pour ou dans le transport des programmes industriels existant vers le composant MC3 doit être étudié. L'ensemble constitue un projet futur en tant que tel. Par contre, il s'agit d'un domaine où l'on dispose en France d'un savoir-faire, des méthodes et des outils remarquables.

### Vers un microprocesseur de traitement de la communication

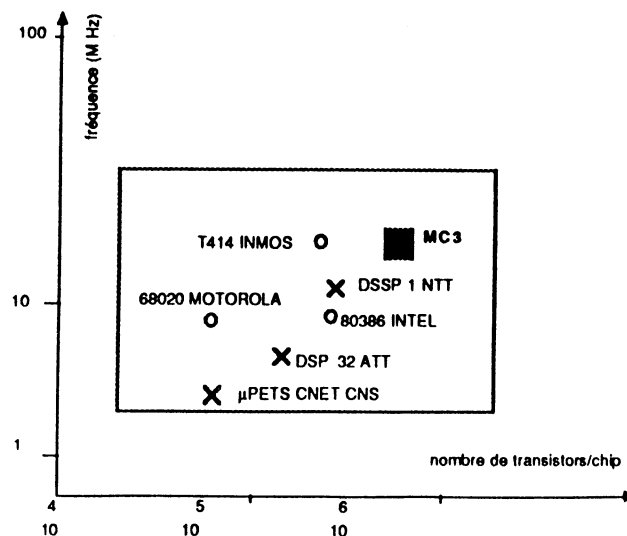
Faisons un parallèle avec les processeurs de traitement digital du signal à usage général dont les études ont longtemps précédé ceux du traitement de la communication. Les plus sophistiqués en terme de performances de ces processeurs, dans l'état actuel de nos connaissances, fonctionnent avec des cycles de base de 50 ns, une technologie CMOS et des opérateurs arithmétiques flottants de 32 bits; leur complexité avoisine les 400.000 transistors [AOY 87].

Le composant MC3 est un *processeur de traitement digital de la communication à usage dédié* aux protocoles de haut niveau cités ci-dessus mais *non fermé*. Il est autant destiné à un protocole de Transport classe 4 de l'ISO qu'à un protocole tel que TCP-IP. Son utilisation pour implanter un protocole type HDLC est a priori facile.

Implanter d'autres protocoles, qui partagent des caractéristiques communes avec celles étudiées

: nécessité de traitement temps-réel, gestion évoluée des interfaces, gestion des trames, etc., revient à étudier les architectures des cartes dans lesquelles se situera le composant, développer les programmes en langage MC3 et les situer sur le bus privé du composant MC3.

MC3 vise pour les années 1990 un cycle de base de 40 ns, une technologie CMOS à 2 métaux d'interconnexion et de pas de grille de  $1\mu 25$ . Sa complexité est en grande partie due aux mémoires internes ROM, RAM et Dual RAM, et aux machines de transfert; elle est estimée, d'ores et déjà, à plus de 750.000 transistors.





## **Chapitre 4**

# METHODOLOGIE DE CONCEPTION DES CIRCUITS DE COMMUNICATION



## 4.1. PRESENTATION DE LA METHODOLOGIE

### 4.11. Introduction

La réalisation d'un circuit intégré comporte deux tâches indépendantes : la conception et la fabrication. Si l'on considère uniquement les facteurs qui interviennent dans le coût de revient dûs à la conception, on s'aperçoit qu'ils dépendent directement de l'expérience et donc de la productivité moyenne du (ou des) concepteurs et des moyens de CAO donnés [BOU 85].

C'est précisément une méthodologie, allant dans le sens de l'augmentation de la productivité et de la sûreté de la conception, que nous avons mise en œuvre.

#### 4.111. POUR QUEL TYPE DE CIRCUIT DE COMMUNICATION

Nous considérons le domaine de la communication de données comme un domaine spécifique, dans lequel les circuits de communication visés par la méthodologie en question sont des *circuits prototypes conçus à la demande* (circuits ASIC). Par circuit prototype, nous entendons un circuit correct au point de vue de la fonctionnalité et des performances, même si la surface utilisée n'est pas optimisée, comme l'est généralement un circuit taillé de manière ad hoc par un concepteur professionnel.

#### 4.112. QUELLE EST LA POPULATION DES PERSONNES VISEES

Il s'agit de la population des *concepteurs dits système*, par opposition à la population des concepteurs professionnels dont le nombre n'augmente pas de manière significative par rapport à celui de la première population. Si, pour simplifier, on considère que la réalisation d'un circuit est la somme du travail d'un concepteur et de la mise à sa disposition d'outils de CAO, et si la compétence du concepteur se déplace plus vers les connaissances système, il faut forcément sophistication des outils de CAO pour maintenir l'équilibre.

D'où la naissance d'une couche de CAO dite "CAO système" qui cherche à capter le savoir faire des concepteurs professionnels pour le mettre à la disposition de tous : ingénieur système, apprenti-concepteur, concepteur confirmé aussi. Le système SCHUSS, Système de Conception Hiérarchisée Utilisable par des Spécialistes Système, développé au département Architecture des Micro-Systèmes du CNET CNS, est typiquement une couche de CAO système.

#### 4.113. LA CAO SYSTEME SCHUSS

Le système SCHUSS offre aux ingénieurs système la possibilité de consulter un catalogue de "composants intégrables", déjà caractérisés. Les fonctions assurées par ces composants doivent être paramétrables pour pouvoir être ajustées aux besoins de l'ingénieur système. Ainsi ces composants



sont-ils appelés blocs flexibles ou blocs paramétrables. Pour permettre la réalisation de ces blocs, la bibliothèque de cellules de base [DOL 88] et le système de programmation de blocs flexibles LOF [BER 86] ont été mis au point. Mentionnons les générateurs existants de RAM, de ROM et les générateurs en cours de développement tels que le chemin de données SPOT, le générateur de placement routage de "standard cells" et les générateurs de modules spécifiques pour les circuits de communication [DAN 88.4].

Le système SCHUSS offre d'autre part des outils de génération automatique, en particulier de de partie contrôle, partant d'un niveau de description du problème dont les ingénieurs système ont l'habitude (niveau d'un graphe par exemple). Mentionnons le générateur automatique de séquenceur GASP [FLA 84], la chaîne ASPR pour Assemblage de PLAs et de ROMs [OLI 87].

SCHUSS s'appuie sur la couche de CAO de VLSI, appelée CASSIOPEE [BEY 82], développée au CNET CNS autour de la base de donnée COSMIC [JUL 87]. Il reste cependant compatible avec d'autres environnements logiciels de conception de circuits intégrés tels que CALMA par exemple.

#### 4.114. DES CIRCUITS DE COMMUNICATION, EN QUELLE TECHNOLOGIE

La technologie de référence est la technologie HCMOS, disponible au CNET CNS. La technologie de base migre beaucoup plus rapidement que la conception des outils ne se met au point. Les règles de connexion sont passées en l'espace de 18 mois du mono-métal en deux métaux. Le pas de grille du transistor diminue (HCMOS1, HCMOS2, HCMOS3).

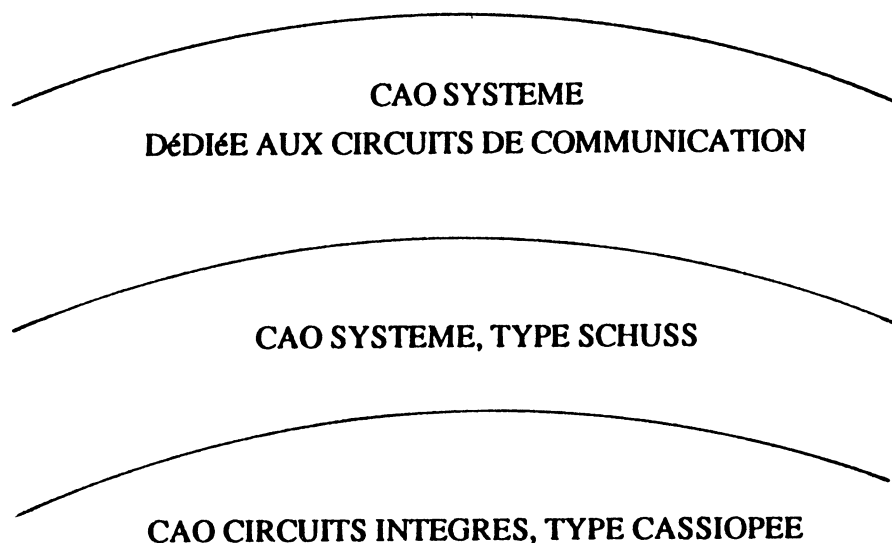
Par conséquent, une équipe de recherche comme la nôtre ne peut guère se permettre d'être trop exposée aux divers changements et de règles technologiques et d'outils de conception. L'expérience accumulée pendant la conception du circuit FIP-VLSI où nous avons mis au point la méthodologie, la bibliothèque de cellules en NMOS L3, en passant par la bibliothèques d'opérateurs flexibles, nous a montré l'importance des couches intermédiaires et des outils associés (simulateurs à plusieurs niveaux, extracteurs, traitement topologique, ...).

#### 4.12. Méthodologie de Conception des Circuits de Communication

Le principe de notre méthodologie est décrit schématiquement dans la figure de la page suivante.

Ce découpage en couches ne sous-entend pas que l'utilisateur final ne voie que l'interface de la CAO système dédiée. Il s'agit en fait d'enrichissement de fonctionnalités de couche en couche, sachant que par ailleurs, l'utilisateur final peut accéder directement aux outils et aux résultats de la couche qu'il souhaite.

## CONCEPTION DES CIRCUITS DE COMMUNICATION



Dans le cas des architectures semi-dédiées, cette méthodologie consiste à adopter la double démarche suivante :

- . une analyse descendante des spécifications du circuit à concevoir; cette analyse permet de définir les principaux éléments (machine, module, opérateur) qui constituent le circuit,
- . la construction du circuit à partir de l'existence d'une bibliothèque de cellules, de la génération des opérateurs et de la construction des modules et des machines.

### 4.121. BIBLIOTHEQUE DE CELLULES

Les parties écrites en italique sont considérées comme des parties déjà existantes ou bien uniquement à spécifier, la réalisation devant être prise en charge par des concepteurs professionnels ou plus tard par des compilateurs de cellules.

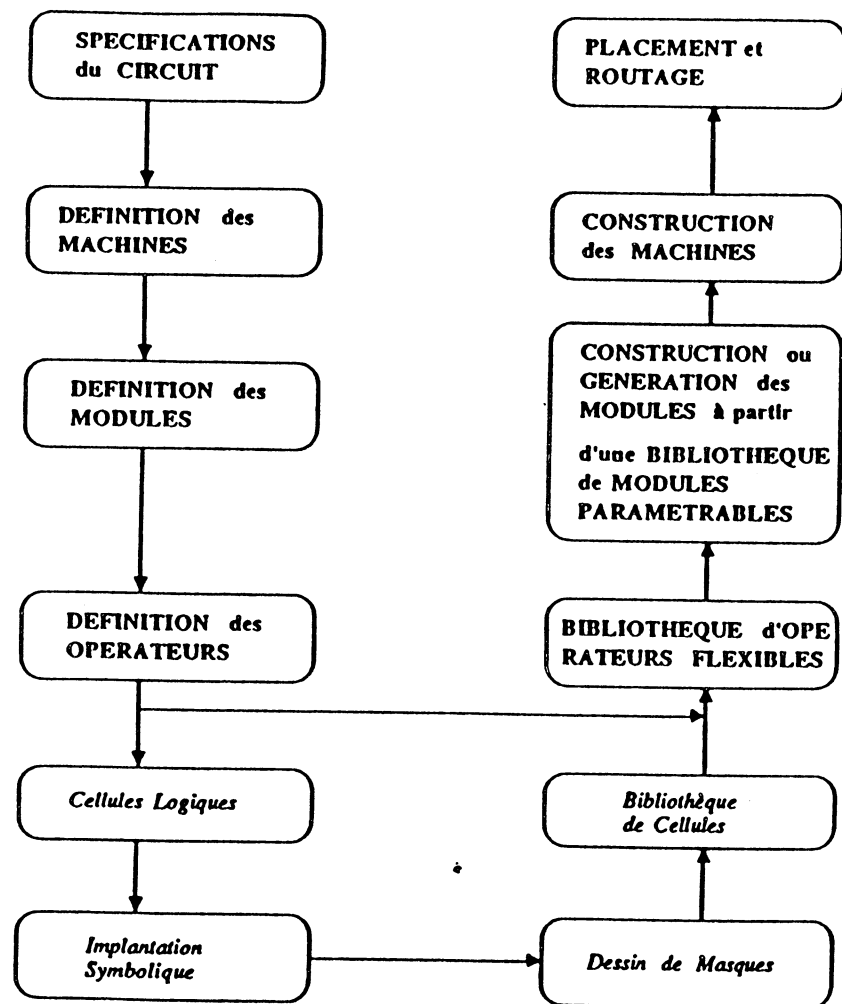
La spécification de chaque cellule nécessite les points suivants.

- . L'enveloppe implantée *qualitative* de la cellule soit définie avec ses points d'entrée-sortie (avec l'orientation de chaque entrée ou sortie et son niveau technologique); par contre sa dimension ainsi que les coordonnées de chaque entrée ou sortie ne sont qu'indicatives. La cellule ne peut

contenir aucun motif réel.

. Le schéma logique de la cellule doit être défini, simulé logiquement.

. Les règles concernant les assemblages potentiels de la cellule avec d'autres cellules doivent être précisées.



Méthodologie de conception

Dans la mesure où la construction générale des opérateurs à partir des cellules est soumise à des règles précises, concernant aussi bien la méthode de construction (par tranche par exemple) que les règles de connexion entre cellules, ces règles peuvent et doivent avoir une influence sur la définition des contraintes sur la cellule (largeur standard, hauteur approximable en fonction de la complexité de la cellule) [DAN 88.4].

#### 4.122. BIBLIOTHEQUE D'OPERATEURS

Un opérateur est un assemblage flexible de cellules de base et réalisant une fonction précise, souvent réductible à une primitive de niveau 'transfert de registre'. La spécification de chaque opérateur entrant dans la bibliothèque, qui de ce fait s'enrichit, doit comporter les points suivants :

- . Les limites de sa flexibilité et la signification de ses paramètres, ainsi que le programme qui sert à la génération de l'opérateur.
- . Son organisation topologique ainsi que la liste des cellules qu'il utilise dans la génération.
- . Les règles d'assemblage et son enveloppe qualitative.

La validation de chaque opérateur généré sera effectuée logiquement. La simulation logique doit s'effectuer, dans le cas idéal, sur des données extraites automatiquement à partir de l'implanté et de la hiérarchie de cellules utilisées.

En ce qui concerne la performance de l'opérateur, elle dépend de celle des cellules qu'il utilise. Si l'on dispose de l'ensemble des cellules requises avec leur performance, la performance de l'opérateur généré peut être évaluée à partir de la simulation électrique des schémas extraits. Cependant, il faut reconnaître que dans ce type de méthodologie, la performance du circuit est le résultat d'une caractérisation et non tout-à-fait un préalable à la conception du dit circuit; cette notion rentre dans la définition du terme prototype mentionné ci-dessus.

#### 4.123. BIBLIOTHEQUE DE MODULES

Le terme module est utilisé ici pour désigner tout assemblage flexible d'opérateurs réalisant une fonction plus complexe et au sein duquel on peut distinguer, selon l'approche à deux niveaux la plus générale, une partie contrôle et une partie opérative (une telle fonction est déjà apparente au niveau de l'architecture fonctionnelle du circuit de communication, comme composant d'un processus ou comme processus lui-même).

La spécification de chaque module doit comporter les points suivants :

- . La définition fonctionnelle du module; une simulation fonctionnelle sera souhaitable.
- . Une étude sur les différentes organisations topologiques du module et sur les flexibilités visées.
- . La spécification des paramètres, de leur signification et de leur limite.
- . L'énoncé des règles d'assemblage des opérateurs.

La partie contrôle du module doit être générée avec les outils de CAO système et simulée. La simulation logique de l'ensemble du module doit être complète entre la partie opérative et la partie contrôle.

#### Notes

La richesse de la bibliothèque de cellules de base (proposée ou non par le fondeur), celle de la bibliothèque d'opérateurs flexibles et de modules paramétrables déjà conçus conditionnent les possibilités d'utilisation d'une telle méthodologie..

Plus on se rapproche de l'utilisateur système, plus les bibliothèques deviennent spécialisées. En particulier, la bibliothèque de modules paramétrables est souvent dédiée à un type de circuits en particulier, pour des raisons d'optimisation et de complétude; cela étant déjà vrai, dans certains cas, pour la bibliothèque d'opérateurs flexibles (le CRC est un bon exemple d'opérateur dédié à la communication).

Un certain abus de langage fait que l'on appelle l'ensemble des connaissances et des programmes qui entourent l'objet module en question, un compilateur de silicium. Les compilateurs qui sont utilisés dans notre méthodologie sont des compilateurs de blocs flexibles et des compilateurs de plan de masse (le compilateur de plan de masse génère une description implantée à partir d'un ensemble d'enveloppes et une liste d'interconnexion à réaliser) [LAR 86].

Pour notre équipe, un objectif à moyen terme sera des compilateurs d'architecture qui, à l'état actuel des choses, posent encore beaucoup de problèmes : un des problèmes majeurs réside dans la description comportementale d'entrée dans le compilateur. Dans le domaine des protocoles de communication, où la description est un domaine largement et précisément débattu, il reste à faire la liaison avec les équipes spécialisées dans le développement de tels langages : par exemple, le système ESTELLE [DIA 87], le système LOTOS [GOD 87].

## 4.2. CONCEPTION DES CIRCUITS DE COMMUNICATION

### TRAVAUX REALISES

La méthodologie décrite a été mise en œuvre pour la conception du circuit FIP-VLSI. Elle a été utilisée pour l'étude de l'architecture de la machine de transfert du composant MC3 et sa validation logique (§ 4. 22).

#### 4.21. Conception du circuit FIP-VLSI [DIA 86.1]

L'architecture fonctionnelle du circuit est décrite dans le paragraphe 2.331. Les différentes fonctions (ou modules) ont été décomposées en opérateurs. La conception de chaque opérateur tient compte des différents types de synthèse préconisée (synthèse des parties opératives en tranches, synthèse en logique aléatoire, synthèse des parties contrôle sous forme de séquenceur).

##### 4.211. SYNTHESE DES MODULES

Le tableau suivant, qui reprend celui définie dans § 2.331, montre comment chaque module a été synthétisée. La dernière colonne concerne plus spécifiquement les paramètres de génération des opérateurs (§ 4.213).

PROCESSUS		FONCTIONS		OPERATEURS	CELLULES	PARAMETRAGE DE GENERATION
INTERFACE STATION		TRANSFERT DES DONNEES	PORT	REGISTRE PARALLELE	1. - BASCULE M-E (A) - SELECTEUR (B) - AMP. INV. MULTIPLEXEUR AMP. 3 ETATS	16 bits  16 (4 vers 1) 16 (plots 3 états)
		SYNCHRONISA- TION DU TRANS- FERT	EMISSION	AUTOMATE D'ETATS FINIS	(A) PLA	2 à 3 bits graphe émission
	RECEPTION		AUTOMATE D'ETATS FINIS	(A) PLA	2 à 3 bits graphe de réception	
	INITIALI- SATION		DEF. DE LA TAIL- LE DU CHEMIN DE DONNEES		ENCODEUR	PLA
		DIRECTION DES DONNEES		REGISTRE DE MEMORISATION	BISTABLE	11 bits
		CHARGEMENT DE L'ADRESSE	AD. PHYSIQUE	REGISTRE // REGISTRE A DECAL. DECODEUR	Identique 1 (A) PLA	4 bits 4 bits équation logique
GESTION DE LA TRAME	EMISSION DE LA TRAME	ENTRAMAGE		SERIALISATEUR	Identique 1 MULTIPLEXEUR	16 bit 1 (4 vers 1)
		GENERATION CRC		GENEREUR CRC	2 - BASCULE M-E avec PA1 - PORTES NON-ET (C) OU-ET (D)	polynôme type
	RECEPTION DE LA TRAME	RECONNAISSAN- CE DE L'ADRES- SE		REGISTRE // REGISTRE A DECAL. COMPARATEUR	Identique 1 (A) 1 - PORTES OU (E), EGALITE (F)	16 bits 4 bits (4 fois) 16 bits
		VERIFICATION CRC		GENEREUR CRC COMPARATEUR	Identique 2  Identique 3	polynôme  reste unique
		DETRAMAGE		DESERIALISATEUR	(A)	17 bits

INTERFACE MEDIUM	METHODE D'ACCES			SEQUENCEUR	PLA (A)	graphe
	EMISSION	GENERATION DE PREAMBULE		COMPTEUR	4 - (A), (B) - Portes : 8 F, NON-OU (2,3,4 portes) (G), INVERSEUR (H)	5 bits
		CODAGE	CODEUR BIPHASE VIOLATION DU CODE	LOG. ALEATOIRE DECOMPTEUR LOG. ALEATOIRE	(A), (D) 5, (A), (B), (D), (G), (H) PORTES ET (I) (E), (F)	structure irrégulière
	RECEPTION	REMISE EN FORME		LOG. ALEATOIRE	(A), (C), (E), (H)	structure irrégulière
		SUPPRESSION DE PREAMBULE		DESERIALISATEUR DECODEUR	(A) PLA	10 bits équation logique
		DECODAGE		REG. A DECALAGE LOGIQUE ALEATOIRE	(A) (A), (D)	5 bits structure irrégulière
SERVICE	GENERATEUR D'HORLOGES		OSCILLATEUR	OSCILLATEUR	OSCILLATEUR	
			DIVISEUR DE FREQ.	Identique 4 générateur horl. non recouvrantes	3bits 5, 10, 20 MHz	

Le tableau montre la répétition dans l'utilisation d'un groupe d'opérateurs pour la construction des modules et d'un groupe de cellules pour la réalisation de ces opérateurs. En conséquence, nous avons proposé dans le cas du circuit FIP-VLSI, la construction d'une bibliothèque d'opérateurs flexibles et d'une bibliothèque de cellules. Ces bibliothèques, qui étaient destinées à des circuits de communication en technologie NMOS, ont été originalement constituées par les éléments identifiés dans le cas du circuit FIP-VLSI.

Le tableau mentionne d'autre part que certains opérateurs possèdent une structure irrégulière ou nécessitent un traitement particulier : violation de code, codeur et décodeur biphasé, ... Ces opérateurs ont fait l'objet d'implantation manuelle.

#### 4.212. CONCEPTION DE LA BIBLIOTHEQUE DE CELLULES

Nous avons défini plusieurs règles pour la réalisation de cette bibliothèque, dans le but d'aider l'implantation en tranche des différents opérateurs, d'optimiser les interconnexions entre cellules et de simplifier son utilisation. Plusieurs de ces règles correspondent aux propositions faites dans [SUZ 81] [REI 83]. Ces règles sont les suivantes :

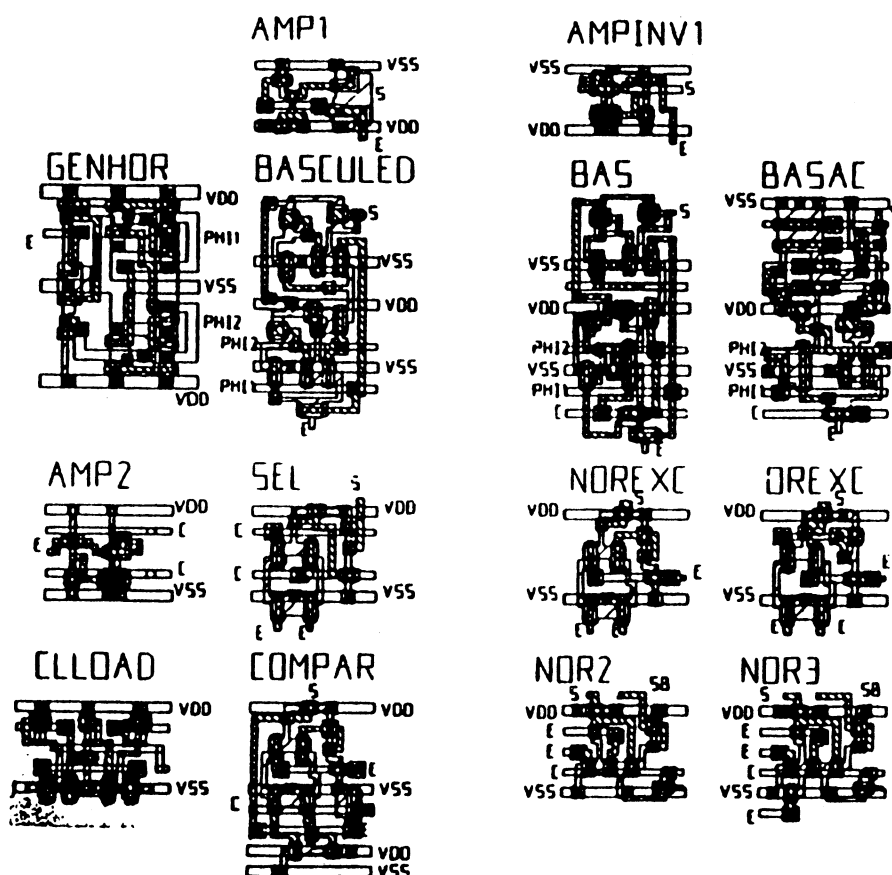
- . Il faut respecter pour toutes les cellules une largeur déterminée.
- . La cellule doit être construite avec des transparences parallèles aux fils d'aluminium (VDD, VSS, horloges et fils de contrôle). Cette structure permet, lors de la construction de l'opérateur, de relier les lignes d'alimentation en formant une structure en peignes imbriqués.
- . Il faut essayer de placer les fils de contrôle, d'alimentation ou d'horloge à la même hauteur afin de faciliter l'assemblage horizontal des cellules.

. Les entrées et les sorties de deux cellules potentiellement adjacentes doivent être du même côté. Cela oblige à réaliser les cellules avec les entrées du côté par exemple inférieur et les sorties du côté supérieur. Il arrive que la seule juxtaposition de deux cellules permette de les relier.

Ces règles sont illustrées par la figure de la page suivante qui présente quelques cellules constituant la bibliothèque :

- . bascule maître-esclave BASCULED, bascule maître-esclave avec remise à zéro synchrone BAS, bascule maître-esclave avec remise à zéro asynchrone BASAC,
- . sélecteur SEL, multiplexeur,
- . inverseur, amplificateur AMP1, AMP2, amplificateur-inverseur AMPINV,
- . portes diverses : ou-exclusif OREXC, égalité COMPAR, NON OU à plusieurs entrées, OU, ET, NON ET à plusieurs entrées, cellule de contrôle CLLOAD
- . générateur d'horloge GENHOR.

La bibliothèque a été réalisée dans une représentation symbolique 'bâton' sous CASSIOPEE appropriée à la technologie NMOS. Les vérifications des règles de dessin, de bon fonctionnement électrique de chacune des cellules ont été effectuées par l'utilisation des divers simulateurs ad hoc. L'implantation au micron des cellules présentée ci-après a été générée automatiquement et vérifiée sur le système CAO CALMA.





**Notes :** Cette bibliothèque a été réalisée dans l'intention de fournir les cellules de base nécessaires à la construction des opérateurs en tranches. Elle ne constitue pas une bibliothèque dans le sens d'une bibliothèque de cellules précaractérisées. Elle n'est pas complète et ne peut prétendre à elle seule être suffisante pour la conception d'un circuit. Dans l'optique de la méthodologie développée ici, la bibliothèque de cellules se doit d'être validée industriellement et d'être multi-fondeurs.

#### 4.213. PROGRAMMATION DE LA BIBLIOTHEQUE D'OPERATEURS FLEXIBLES

Il s'agit d'opérateurs conçus en tranches de bits. Ces tranches, construites par éléments ou cellules juxtaposées les uns aux autres, peuvent être répétées  $n$  fois selon les caractéristiques de l'opérateur.

L'utilisation des outils de CAO du type 'assembleur de silicium' offre la possibilité de construire ces opérateurs de manière flexible (programmée) et par conséquent de définir leurs caractéristiques à partir d'un ensemble de paramètres.

Cette bibliothèque a été définie à partir de la bibliothèque de cellules, selon les critères suivants :

- . A partir du schéma logique, chercher la régularité et la modularité de chacun des opérateurs pour pouvoir construire des opérateurs en tranche.

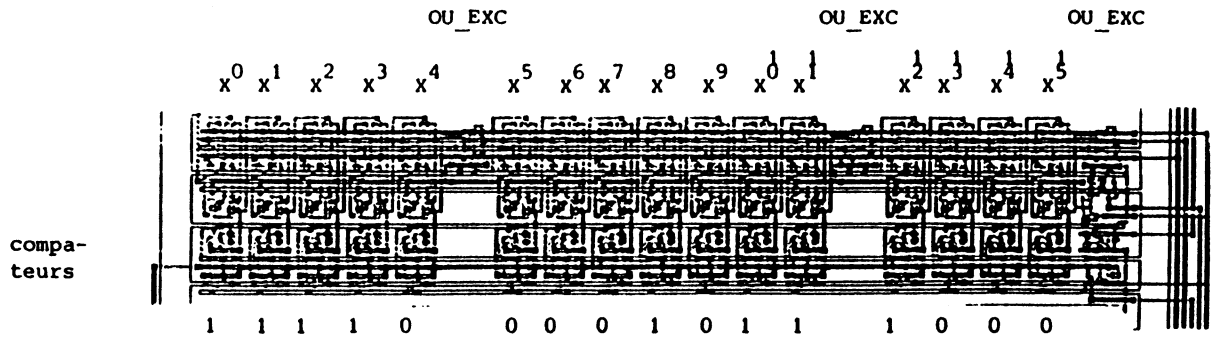
- . Définir les possibles paramètres à utiliser pour réaliser de manière automatique un ensemble d'opérateurs : par exemple, le nombre d'éléments binaires de l'opérateur, l'écartement entre éléments binaires, position des entrées et des sorties de l'opérateur, .... Dans le cas typique du CRC, par exemple, il s'agit de définir le polynôme générateur et le type de reste à détecter (polynôme caractéristique ou reste nul).

Chaque opérateur est donc le résultat de l'exécution d'un programme, écrit dans le langage LOF (Langage d'Opérateurs Flexibles) [BER 86]. Ce langage permet la manipulation des cellules ou des blocs, contenus en l'occurrence dans la base CASSIOPEE. Il peut effectuer des opérations sur un seul objet (symétrie, rotation, répétition, ...) ou entre deux objets (assemblage par aboutement horizontal, vertical, ...).

Les opérateurs obtenus par cette méthode, dans le cadre du circuit FIP-VLSI ont été les suivants : registres à décalage, sérialisateurs, désérialisateurs, registres parallèles, compteurs, décompteurs, générateur / vérificateur de CRC.

La figure suivante donne l'exemple d'un schéma implanté obtenu par appel au programme de génération de l'opérateur CRC avec pour paramètres : le polynôme du CCITT.

OPÉRATEUR CRC, généré automatiquement avec les paramètres suivants :  
 POLYNOME GÉNÉRATEUR  $x^{16} + x^{12} + x^5 + 1$  ( avis V-41 CCITT)  
 RECONNAISSANCE DU POLYNOME CARACTÉRISTIQUE



#### 4.214. CONSTRUCTION ASCENDANTE DU CIRCUIT FIP-VLSI

La deuxième partie de la méthodologie proposée est la construction ascendante du circuit. Elle consiste à :

- . implanter les différents modules à partir de la bibliothèque d'opérateurs flexibles et de la bibliothèque de cellules,
- . placer les modules et router les interconnexions pour former le circuit,
- . effectuer les diverses vérifications de bon fonctionnement avant la fabrication du circuit.

Le placement et le routage de l'ensemble du circuit FIP-VLSI ont été effectués de manière manuelle, en l'absence des outils adéquats. Les commentaires qui accompagnent la photographie du circuit donnent quelques indications sur la nature des modules et des opérateurs du circuit FIP-VLSI.

La conception de la maquette du circuit FIP-VLSI a été effectuée en technologie NMOS L3. Suite à l'arrêt de la filière NMOS L3 au CNET CNS, il n'y a eu qu'un "run" dans la réalisation du circuit.

#### 4.215. CONCLUSIONS

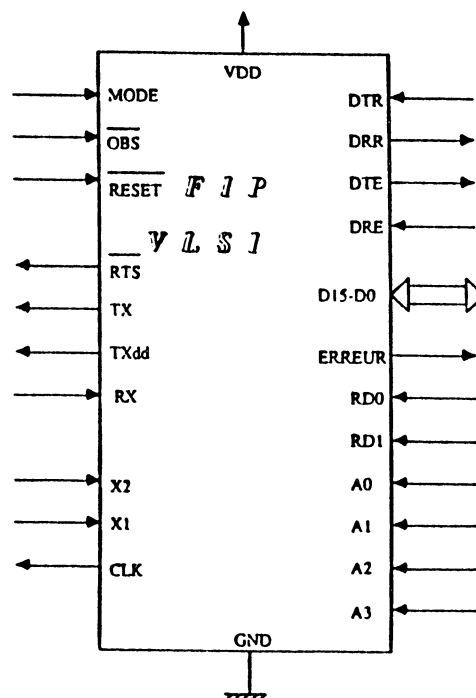
Un des enseignements que nous avons pu tirer lors de cette expérience de réalisation d'un circuit à la demande à partir d'une bibliothèque d'opérateurs flexibles est la diminution considérable du temps de conception. Il s'avérait impératif par contre d'avoir des outils pour le placement et le routage automatiques des modules, placement et routage qui ont été faits, dans notre cas, à la main ou obtenus comme résultats d'une programmation. La grande surface occupée par le premier "run"

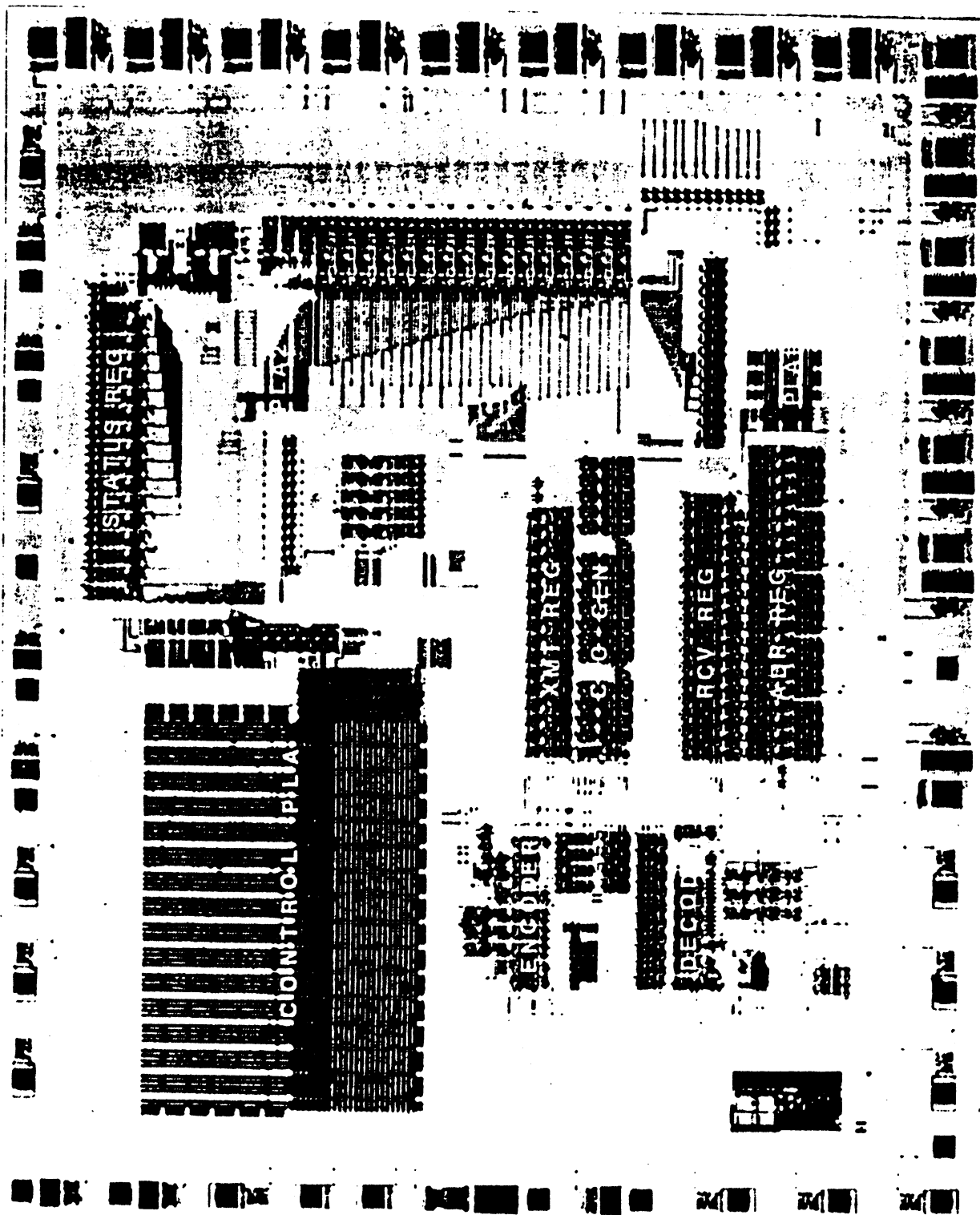
circuit (35 mm<sup>2</sup>, consommation d'environ 500 mW) provient en partie de ce manque d'outils de CAO de placement-routage.

Notre souci principal concernait, au demeurant, la méthodologie; elle devait permettre de définir les modules fonctionnels caractéristiques et de les répertorier dans une bibliothèque. L'exemple du circuit FIP a mis en relief la construction d'une bibliothèque d'opérateurs flexibles et les opérateurs qui peuvent la constituer.

Un des défauts de la méthodologie telle qu'elle a été appliquée réside dans la nécessité de ladite bibliothèque de cellules. Bibliothèque de cellules que nous avons été amenés à concevoir pour notre propre usage, sans en être des spécialistes, avec pour conséquence un manque de généralité dans la définition des entrées-sorties, un taux moyen de compacité, une sous-utilisation des transparences, etc. D'autre part, à chaque changement de technologie, cette bibliothèque est à refaire. Ces conclusions nous ont amené à nous situer depuis lors en amont des problèmes de bibliothèque de cellules [DAN 88.4]. Nous considérons que les concepteurs système doivent pouvoir à la conception d'un circuit se baser sur une bibliothèque de cellules conçues dans un cadre industriel, avec une stratégie de routage donnée et connue [CIB 88]. A défaut, si une cellule manque ou si elle ne peut être obtenue à partir d'autres cellules, le concepteur système ne peut et ne doit que spécifier son environnement d'utilisation et son enveloppe qualitative.

Le circuit FIP-VLSI n'est pas un circuit complexe. Sa complexité s'élève à environ 15.000 transistors équivalents. Il est cependant caractéristique de ce type de circuits de communication dont le savoir-faire aussi bien en architecture qu'en conception est loin de celui qui existe dans les domaines des microprocesseurs ou du traitement du signal.





Le circuit FIP-VLSI

#### 4.22. Définition et Validation de l'architecture de MC3

A partir des spécifications du circuit MC3 et de la définition de son architecture fonctionnelle (§ 3), l'ensemble des architectes ont adopté la double démarche suivante pour valider l'ensemble de l'architecture du circuit :

- . une étape de simulation comportementale et fonctionnelle de l'ensemble du circuit MC3 et
- . une étape de simulation logique, jusqu'au niveau 'portes', de l'architecture logique de chaque processeur du circuit.

##### 4.221. SIMULATIONS COMPORTEMENTALE ET FONCTIONNELLE VERSUS EMULATIONS

Une approche classique de validation de l'architecture d'un circuit de moyenne complexité consiste à faire :

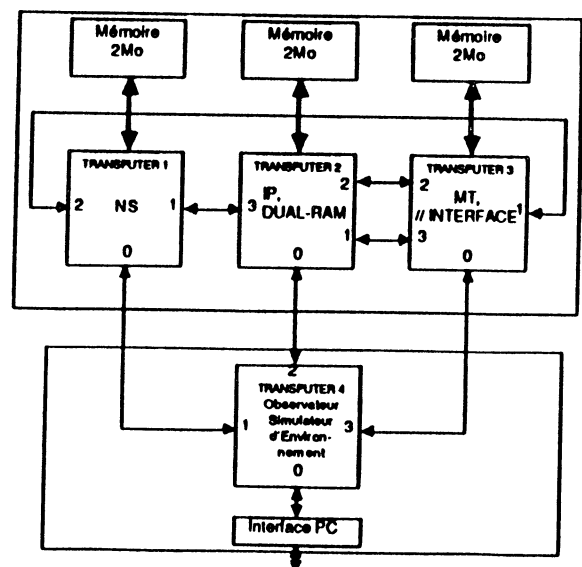
- . l'émulation du circuit en composants discrets pour permettre de valider son architecture interne et pour pouvoir remplacer le circuit dans l'environnement système où il doit être utilisé et
- . éventuellement l'émulation de l'environnement futur du circuit pour permettre de tester le circuit.

En fait, cette approche se révèle relativement inopérante pour des circuits complexes (coût et délai de réalisation élevés, dérive entre les spécifications de l'émulateur et les spécifications du circuit, ...).

Pour le circuit MC3, la démarche adoptée a été la suivante :

##### A. Simulation comportementale

A partir du modèle d'architecture retenu, la machine MC3 est implantée de manière logicielle, avec le langage OCCAM sur un réseau de TRANSPUTERS.



Chaque TRANSPUTER simule le comportement d'un processeur de MC3 : l'IP, le noyau NS et la machine MT. Chaque TRANSPUTER a son propre espace mémoire. Les mécanismes de communication et de synchronisation ainsi que l'exécution parallèle des processeurs de MC3 sont respectés par la simulation comportementale.

Le but principal de cette simulation du comportement de l'architecture de MC3 est de réaliser un outil qui permettra dans le contexte d'une mini-application de

- . valider l'exécution correcte des instructions de l'IP et de NS et
- . vérifier le séquençement entre les machines câblées de MC3.

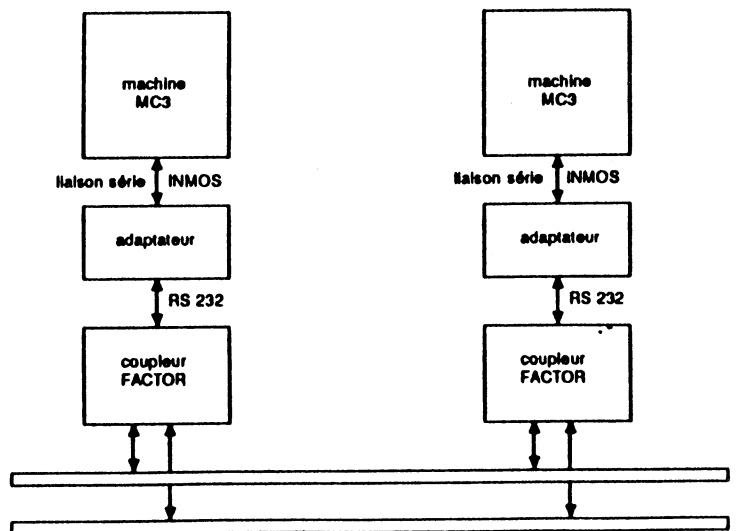
## B. Simulation fonctionnelle

A partir de la maquette du simulateur comportemental du circuit MC3, une simulation fonctionnelle a été effectuée. Elle a été bâtie à partir de :

- . d'une mini-application considérée comme représentative, implantant un protocole de Transport ISO, classe 0 (avec des fonctionnalités de contrôle d'erreur de la classe 1),

- . d'un support de communication pour la mini-application, en l'occurrence un réseau local industriel FACTOR [APT 87.1] et

- . d'une maquette de compilateur de langage externe OCCAM vers des instructions de la machine cible MC3 complète [RAR 88].



Cette simulation fonctionnelle, qui utilise la maquette de simulation comportementale, a permis de :

- . valider le jeu d'instructions choisi pour IP et NS : par instruction et par occurrence de groupe caractéristique d'instructions,
- . de mesurer l'efficacité du parallélisme des processeurs dans MC3 et
- . d'évaluer les performances.

Note : Le temps d'exécution d'un module logiciel en langage MC3 (module obtenu par compilation) est calculé en fonction du nombre de cycles de la machine MC3 (mentionnons que les outils ont été développés pour le rendre calculable [MC3 88.3]) : à partir du temps de base d'un cycle du circuit final (estimé en fonction d'une technologie donnée), on évalue le temps d'exécution du module.

### C. Résultats remarquables

L'analyse des résultats des deux simulations est développée dans [DIO 88]. Elle porte sur plusieurs points :

- . *Le débit de la couche observée* , hors machine de transfert, est évalué à **4000 trames/seconde**, i.e. de 8,2 Mbits/seconde avec des trames de longueur moyenne de 256 octets, malgré les différentes contraintes non favorables à une étude de performances qui sont dues à des hypothèses restrictives d'implantation.

- . Le débit d'une machine de transfert est de **16 M bits/seconde**. Cette valeur est confirmée par les résultats extraits de la simulation logique de l'architecture de la machine de transfert (§ 4.222).

D'autres observations numériques ont pu avoir lieu [MC3 88.3], [DIO 88], concernant :

- . la variation du débit des machines de transfert en fonction du nombre de tampons de données utilisés dans la structure de liste;
- . la variation du même débit en fonction de la taille des trames;
- . la variation du débit des machines de transfert en fonction de la taille du FIFO et de la taille des trames;
- . le taux d'occupation (par rapport au temps total) de la gestion mémoire (gestion des listes, gestion des PnPDU) et
- . le taux d'occupation des processeurs IP et NS.

#### 4.222. VALIDATION DE L'ARCHITECTURE PAR SIMULATION LOGIQUE

A partir des spécifications du circuit MC3 et de la définition de son architecture fonctionnelle (§ 3), l'ensemble des architectes de MC3 ont défini de manière hiérarchique les différents niveaux qui composent le circuit : niveau bloc fonctionnel, niveau module, niveau opérateur et niveau primitive. Une bibliothèque d'opérateurs (environ 75 opérateurs flexibles) et une bibliothèque de cellules (environ 50 cellules) ont été définies, les opérateurs et cellules étant simulés logiquement.

##### Stratégie de simulation

Comme veut la méthodologie définie, la définition de l'architecture logique des machines de transfert se fait donc de manière descendante. Par contre la simulation logique, qui permet de valider l'architecture retenue, se fait de manière ascendante :

- . Simulation logique de toutes les cellules utilisées.
- . Simulation logique de tous les opérateurs utilisés.
- . La simulation logique de chaque module des machines de transfert est effectuée d'abord sur la partie opérative, puis sur la partie contrôle séparément, avant la simulation totale du module; les parties contrôle sont généralement implantées en architecture mono-PLA.

Le bon fonctionnement de chaque module est constaté lors la simulation logique globale des machines de transfert. Seuls les deux modules les plus complexes des machines de transfert (les deux DMA) ont fait l'objet de simulations complémentaires portant sur des exemples académiques exhaustifs [MC3 88.3] afin de pouvoir examiner les cas de figure les plus complexes au niveau des structures de données de support (§ 3.4).

La simulation logique d'une machine de transfert (émission ou réception) est effectuée de manière globale, c'est-à-dire par flots de données.

Le langage de simulation retenue est HILO3 et le simulateur LUCKY-LOG [APT 87.3].

##### Conclusion

La simulation logique des machines de transfert est l'étape principale de la validation d'architecture. Cette validation étant faite, que peut-on tirer comme autre enseignement de cette lourde étape de simulation ? Elle concerne une critique a posteriori des choix qui nous ont guidé dans la définition architecturale des machines de transfert. Nous avons fait les choix suivants :

- . La machine de transfert est conçue comme un ensemble complexe d'unités asynchrones et communiquant. Il s'agit de l'ensemble le plus complexe de MC3, ne serait-ce qu'en terme de portes logiques. Son architecture semi-dédiée n'est pas régulière et les connaissances de conception ne



sont pas aussi bien maîtrisées que dans le cas d'un cœur de microprocesseur par exemple.

. Les choix de multiplexage des ressources sont réduits, dans une première approche, à leur plus simple expression.

La seule contrainte prise en compte lors de la définition de l'architecture concerne l'adéquation entre le débit de la ligne série (en émission et en réception) et le débit de la machine de transfert correspondante.

Observons quelques performances, compte tenu d'une horloge interne de MC3 à 20 MH et d'une machine INTS ou INTP idéale, c'est-à-dire ayant une réaction immédiate à un top d'horloge près :

. Dans le cas d'une machine réception série, soit une trame arrivant sur la ligne série INMOS et devant être rangée dans un en-tête de descripteur de trame et dans un tampon décrit par un descripteur de tampon; le *débit de la machine de transfert réception* est observé égal à **19,39 Mbits/s**.

. Dans le cas d'une machine d'émission série, soit une trame décrite par un PnPDU de niveau Session, un PnPDU de niveau Transport et deux PnPDU de niveau Réseau (i.e. se trouvant dans deux tampons différents), le *débit de la machine de transfert émission série* est observé égal à **18,64 Mbits/s**.

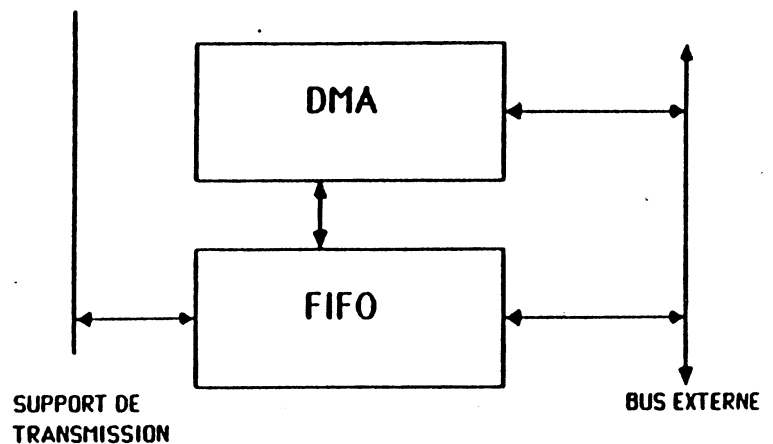
Rappelons que le débit nominal d'une ligne série INMOS est de 14,54 Mbits/s.

### 4.3. ETUDE D'UN COMPILATEUR DE BLOCS FLEXIBLES DMA INTELLIGENT

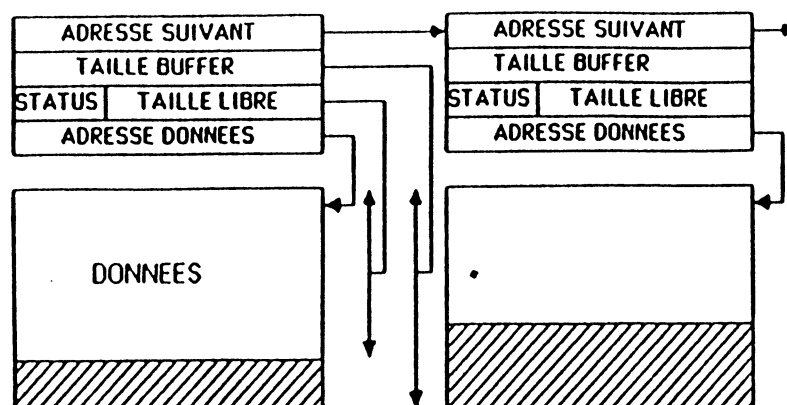
Dans le cadre des compilateurs de silicium de modules flexibles, qui utilisent des outils de CAO-Système tels que assembleur de silicium, générateur automatique de partie contrôle et routeur automatique,

et avec un parti pris de ne spécifier que les cellules de base (sans les concevoir comme dans le cas du circuit FIP-VLSI) et de les considérer soit comme existantes dans une bibliothèque, soit comme réalisables par des concepteurs professionnels soit comme généralisables dans le futur par des compilateurs appropriés,

nous avons implanté *un générateur de DMA intelligent* (ou un compilateur de bloc flexible DMA) en technologie CMOS avec un niveau de métal pour les interconnexions [DAN 88.3]. Ce DMA, outre le fait d'assurer les transferts du FIFO vers une RAM et vice versa, prend en compte une organisation de la mémoire externe structurée en liste de descripteurs de trame chaînés, chaque descripteur de trame pointant sur une liste de descripteurs de tampon chaînés et chaque descripteur de tampon pointant sur un tampon libre.



La figure suivante donne la définition de la structure des descripteurs de tampon d'une trame donnée. Cette organisation de la mémoire externe est à gérer par le système d'exploitation de la station microprocesseur qui est interfacée avec le réseau par l'intermédiaire du circuit de communication.

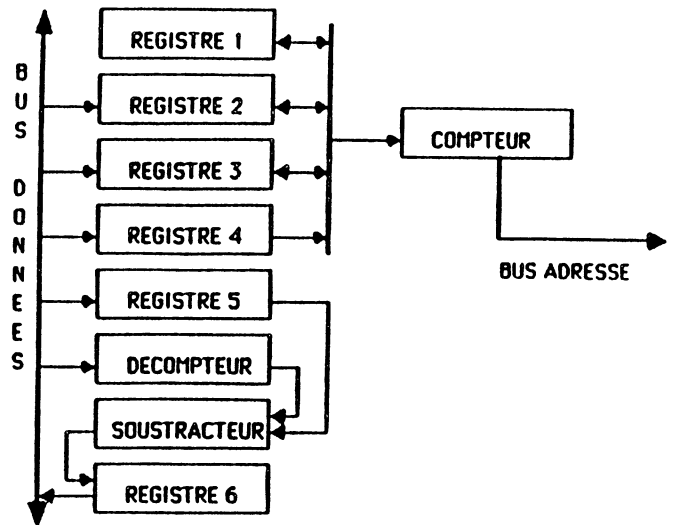


Le schéma logique de la partie opérative du DMA comporte deux unités :

- . l'unité de comptage ayant 4 registres (@ du premier descripteur, @ du descripteur de tampon courant, @ du prochain descripteur de tampon et @ du tampon de donnée) ainsi qu'un compteur, et

- . l'unité de décomptage ayant 2 registres (taille libre, taille utilisée), un décompteur et un soustracteur.

Nous avons choisi de faire la synthèse de la partie opérative en tranche et la synthèse de la partie contrôle sous forme de mono-PLA après l'évaluation de diverses architectures de partie contrôle.

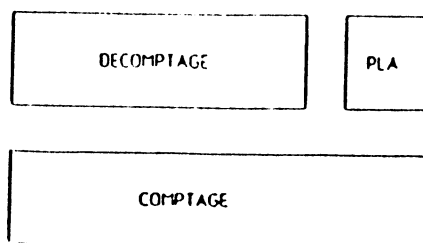


La flexibilité du générateur tient essentiellement aux paramètres suivants :

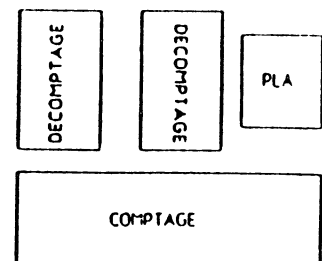
- . prise en compte des différentes tailles du bus d'adresse externe (de 8 à 32 bits),
- . prise en compte des différentes tailles du bus de données externe (de 8 à 32 bits).

Les choix de placements, effectués au préalable, ont induit, selon certaines configurations de paramétrage, des choix de *pliage d'opérateurs* et de routage complexes.

32 BITS



16 BITS



8 BITS

IDENTIQUE AU 16 BITS AVEC UN MULTIPLEXAGE

Hormis la structure de liste orientée circuits de communication, les autres hypothèses de travail, prises en compte dans l'écriture du générateur, concernent l'interface interne du DMA généré :

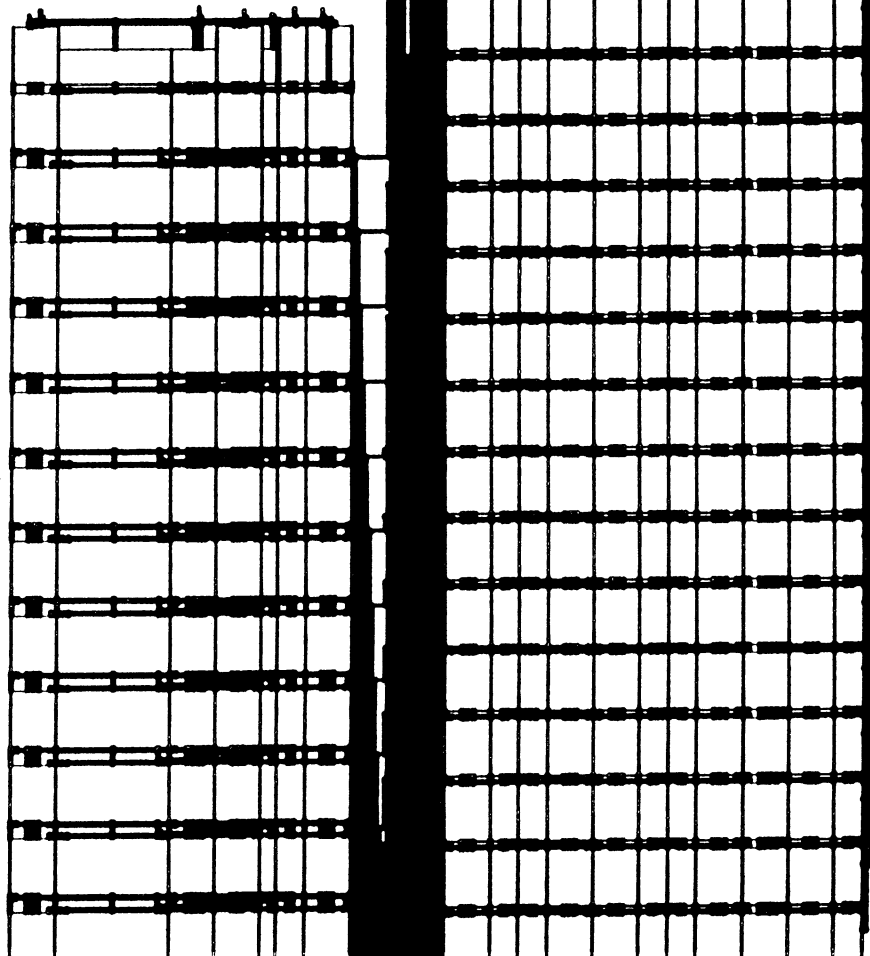
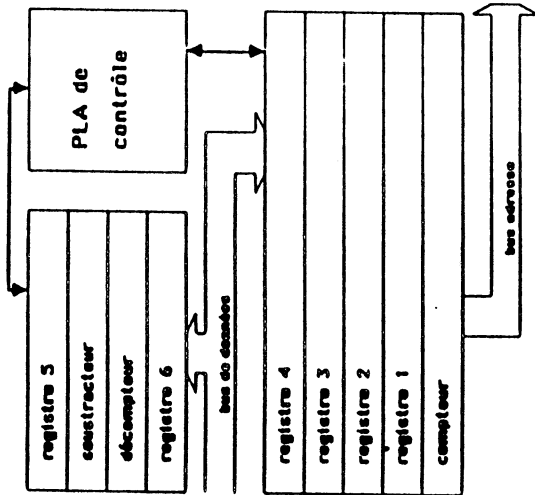
- . règle unique pour l'interface de commande,
- . interface en amont imposée avec la machine de réception par FIFO paramétrable; la paramétrisation du FIFO permet de régler les tailles de données pour rendre le couple FIFO et DMA homogène et de moduler son seuil de vidage (fixé éventuellement par défaut) et
- . interface en aval imposée avec l'unité de gestion d'accès au bus microprocesseur.

Les outils de CAO-Système utilisés pour réaliser le générateur sont ceux du projet SCHUSS au CNET CNS, en particulier LOF v.3; il s'agit du programme LOF v.3 le plus complexe (plus de 3000 lignes) écrit dans l'état actuel de nos connaissances. La simulation fonctionnelle est réalisée en langage de programmation de haut niveau classique. La simulation logique a été faite en EPILOG sur des choix de paramétrages courants.

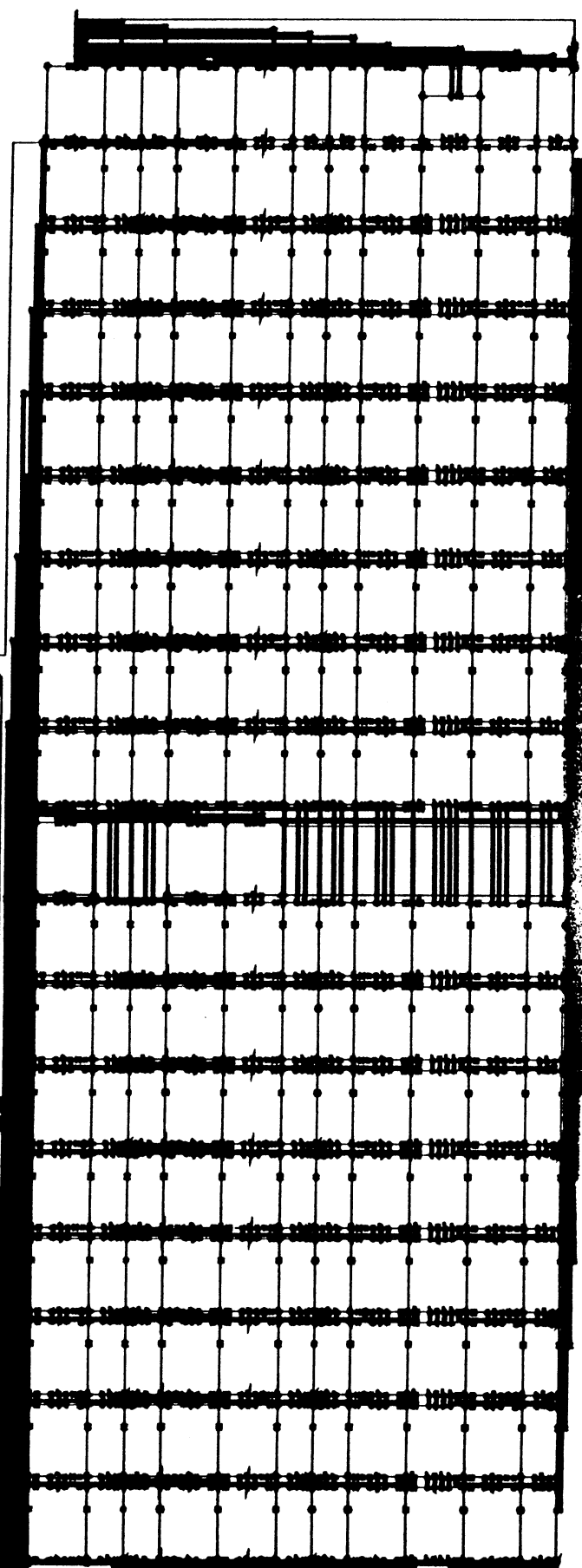
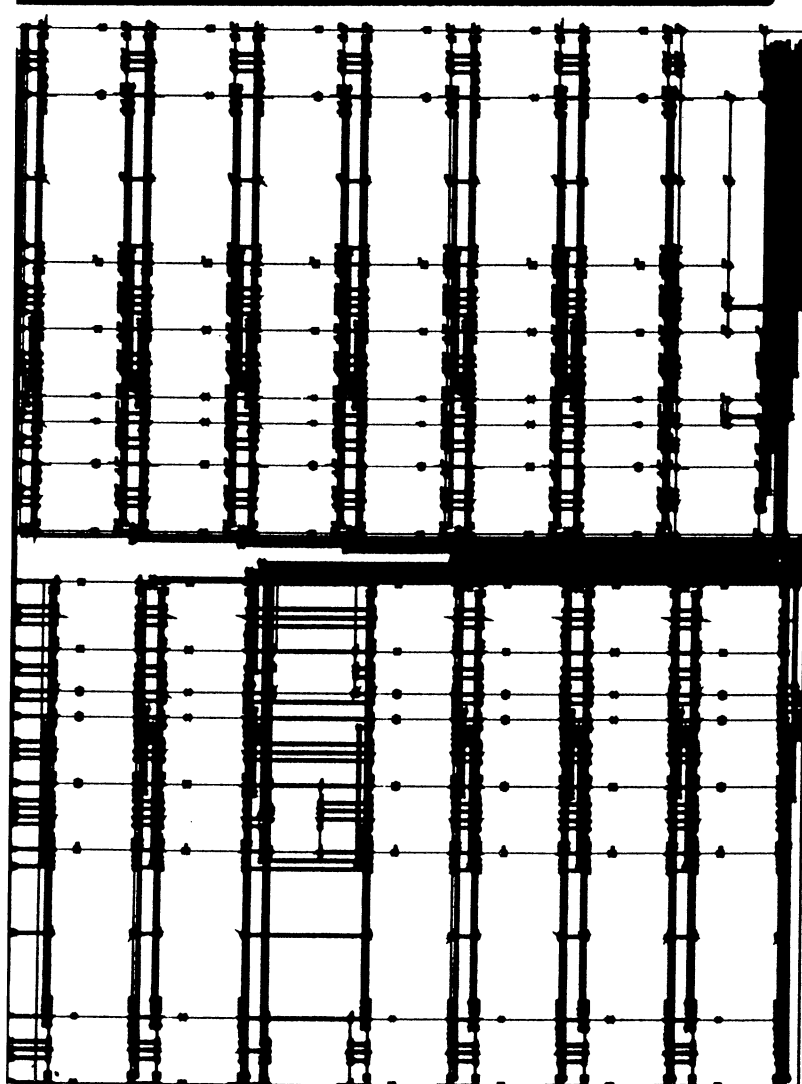
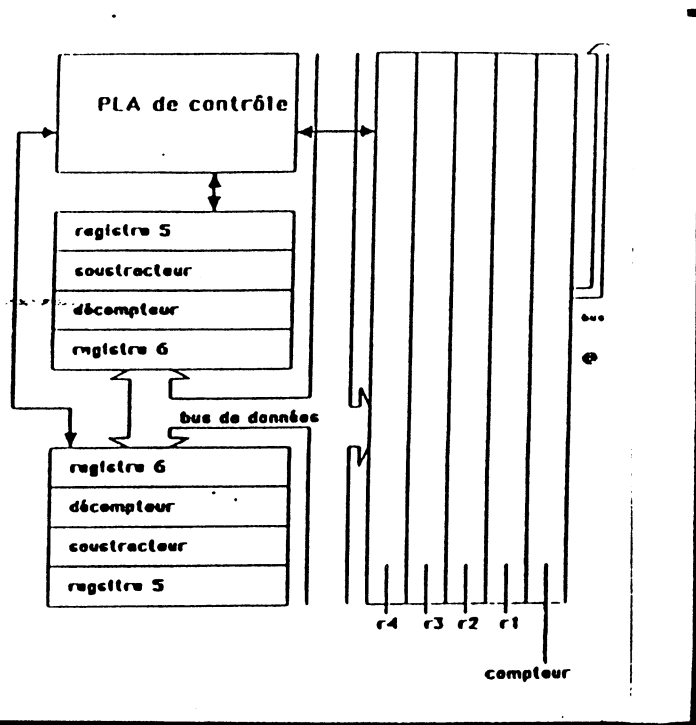
Les figures des pages suivantes illustrent les résultats de deux générations de DMA avec des paramétrages différents. Les rectangles manquants représentent à l'échelle les parties contrôle de ces DMA que nous n'avons pas jugé utile d'incorporer dans le texte.

Note : Ce générateur a été réalisé partiellement dans le cadre d'une convention de recherche effectuée pour le projet SCORE de l'INRIA (§ 2. 332). Il avait précédé dans le temps la définition et la validation de l'architecture de la machine de transfert qui a repris pour son DMA d'interface avec le microprocesseur (dit DMA structure de liste) les idées développées dans ce générateur de DMA, avec des différences dues essentiellement à la définition des descripteurs de trames.

DNA 32 bits ( données : 32 bits  
adresse : 32 bits  
décompteur : 12 bits



DMA 8 bits ( données : 8 bits  
 ← 16 bits  
 décompteur : 12 bits

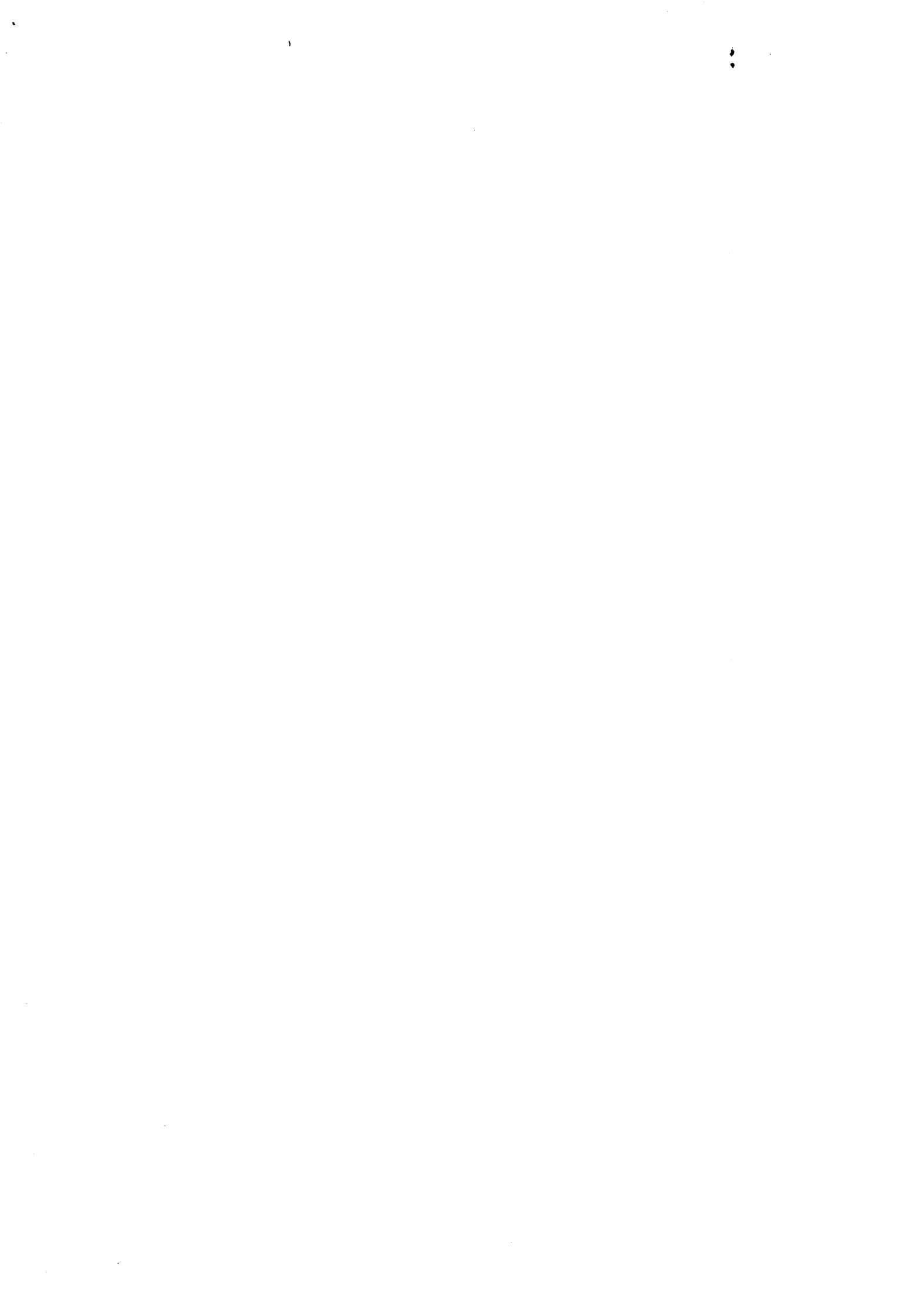




## Chapitre 5

### CONCLUSION





Le domaine des réseaux, qu'ils soient généraux ou locaux, migre sans cesse vers des exigences de débit de plus en plus élevé (l'état de l'art de la technologie de la transmission par fibres optiques en témoigne) cumulées avec des exigences de confort d'utilisation de plus en plus sophistiquées (soit en terme de spécificité comme le temps-réel, soit en terme de fonctionnalité, comme le besoin des couches hautes de l'ISO ou des couches équivalentes, telles TCP-IP). Pour cette nouvelle génération des réseaux et pour le RNIS, l'existence de circuits VLSI et des cartes de communication à hautes performances est primordiale pour le traitement des protocoles.

Pour suivre cette évolution, la technologie des VLSI (filière sub-micronique, bi-CMOS, GaAs) va jouer un rôle primordial dans l'augmentation des performances et l'ajout des fonctionnalités aux systèmes existants.

Les progrès en *architecture et en méthodes et outils pour la conception des circuits ASIC de communication* participent à cette évolution générale, afin de contribuer à abaisser les coûts de développement d'une grande variété de circuits VLSI de communication, y compris ceux prévus avec un volume moyen de production.

Pour assurer une certaine continuité dans les axes de recherche possibles de notre équipe, nous présentons ci-après deux domaines et deux projets auxquels le savoir-faire et l'envie d'explorer le sujet prédisposent particulièrement notre équipe.

## **5.1. VERS UNE MACHINE OUVERTE A HAUTES PERFORMANCES POUR LE TRAITEMENT DES PROTOCOLES DE COMMUNICATION DIGITALE**

A partir des résultats de l'étude MC3 et en faisant l'hypothèse d'une architecture multiprocesseurs, un axe de recherche possible porte sur la définition et la conception d'une telle machine. Les débits visés pourront être de l'ordre de 10.000 trames/seconde pour des trames de taille moyenne de 256 octets. Cet axe de recherche se développerait de la manière suivante.

### **5.11. Extensions de possibilités fonctionnelles**

La définition d'un modèle d'architecture tel que la machine ouverte de communication doit pouvoir permettre aux produits d'évoluer facilement en terme d'extension de possibilités fonctionnelles et en terme d'intégration de sous-ensembles existants. De ce fait, il s'agira d'étudier :

A. l'adaptabilité de la nouvelle architecture et de ses processeurs aux normes et standards suivants : GAM T103, TCP-IP, MAP, CNMA, niveau 3 de traitement des données du RNIS;

B. l'extensibilité de cette architecture et de ses processeurs vers les niveaux 2 : HDLC, LLC et MAC de IEEE 802, X25.2 du CCITT, niveau 2 du traitement de données du RNIS et

C. la prise en compte de l'aspect temps-réel dans l'architecture de cette machine.

Dans une deuxième phase, les différents points critiques observés dans le modèle d'architecture de MC3 feront l'objet soit d'optimisation, soit d'introduction de nouvelles solutions.

D. Un des processeurs de la machine sera un microprocesseur, dont les instructions sont orientées communication. L'ensemble d'instructions qui a été défini lors de l'étude MC3 nécessite, au vu des résultats de simulation, une remise en cause partielle en terme d'architecture interne du microprocesseur et d'architecture de la machine de communication. Cet affinement concerne en particulier le parallélisme entre le fonctionnement du microprocesseur et celui de l'unité de contrôle central qui le gère. Un autre aspect, que ce projet peut prendre en compte, est le suivant : nous voulons mettre à l'épreuve le choix des instructions du microprocesseur à partir des contraintes propres à la génération du code dans la compilation d'un langage externe choisi. Cet aspect (de synergie entre les architectes et les spécialistes de compilation) doit être mené en collaboration avec l'équipe qui gèrera l'aspect logiciel de ce projet.

E. Notion de processeur d'interface : Outre des notions qui ont été développés dans le projet MC3 (interfaces, structures de données, communication et partage de ressources avec les autres processeurs) qui doivent être optimisées ou remises en cause, les nouvelles notions suivantes seront étudiées :

- paramétrisation des processeurs d'interface,
- mémoire-cache,
- configuration dynamique du réseau de processeurs d'interface.

Les modèles d'architectures obtenus feront l'objet de la même démarche de simulation que pour MC3 : simulation comportementale et simulation fonctionnelle.

## 5.12. Conception de la machine et des outils de développements

Dans une deuxième phase de développement, le nécessaire partenariat industriel permettra d'envisager :

A. L'intégration de la machine sous la forme d'un circuit ou de plusieurs circuits, sachant que la notion de machine de transfert fera l'objet soit d'un module spécifique de la

conception, soit d'un circuit dédié. Une implantation dégradée de la machine est envisageable avec un composant microprocesseur du commerce (pour ne pas nommer le TRANSPUTER) à la place de l'Interpréteur de Process, un automate pour le Noyau Système et des composants Machine de Transfert.

B. La définition et la réalisation d'outils d'aide au transport des logiciels industriels existants dans le domaine des protocoles de communication. La capacité de prouver la transportabilité des logiciels existants (machine virtuelle de gestion des processus, langage de haut niveau pour l'exécution du code séquentiel, structure de données spécifique) vers la machine ouverte de communication conditionne l'accueil que le projet trouvera au niveau des bureaux d'études.

L'ensemble des résultats, simulateurs et modèles compris, doit former une structure d'accueil, ensemble de modèles matériels et logiciels de base, utilisables pour la validation de nouvelles applications ou pour la réalisation de nouveaux sous-ensembles du système.

### **5.13. Aspects théoriques**

Les aspects théoriques abordés dans un tel axe de recherche pourront porter sur les sujets suivants :

- . La nature du parallélisme dans l'exécution d'une couche de protocole de communication et dans l'exécution de plusieurs couches adjacentes.

- . La complexité des structures de données, des algorithmes associés et de leur placage sur des architectures matérielles.

- . De l'utilisation d'OCCAM et du TRANSPUTER comme langage, système et machine de modélisation des éléments des protocoles de communication, en vue non point des études de vérification ou de spécification des protocoles, mais de la prédiction des performances des machines implantant les dits protocoles.

## 5.2. EXPLORATION DU DOMAINE DE LA COMPILATION D'ARCHITECTURES DEDIEES AUX CIRCUITS DE COMMUNICATION

### 5.21. Motivations

L'étude des architectures des circuits de communication a consisté pour nous, dans une première approche, à spécifier et à réaliser des exemples d'architectures qui permettent l'implantation de divers protocoles de communication de complexité différente et croissante (étude du circuit FIP proche du niveau physique; étude de l'architecture du circuit CSMA/RCD du niveau MAC; étude de l'architecture du circuit MC3 pour les couches 3,4,5 ISO; étude de l'architecture de la machine de transfert du circuit MC3).

Dans une deuxième étape, (qui a commencé en 1988 et qui se terminera fin 1989) nous avons spécifié et nous réalisons des générateurs de blocs flexibles spécialisés, appelés par ailleurs compilateurs de silicium, qui permettent d'accélérer la conception de circuits intégrés de communication. Parmi ces générateurs, mentionnons *celui qui génère automatiquement l'architecture matérielle, appelée hacheur, susceptible de reconnaître une trame en réception et de la traiter* ; et ce en fonction de la grammaire de la trame d'entrée [DAN 88.4]. Ce travail est issu de nos réflexions sur les codes-fonctions, mais n'a pas fait l'objet de développement dans cet exposé pour des raisons d'homogénéité de l'ensemble du texte.

Il s'agira dans une troisième phase de chercher à proposer des architectures optimisées en fonction des contraintes (fonctionnelles, système, environnementales).

Le passage d'aval en amont du "faire", qui permet d'accumuler des connaissances sur des exemples précis, au "raisonner", qui permettra d'exploiter en quelque sorte cette base de connaissances, n'est pas propre à notre équipe, mais participe d'une tendance générale qui se manifeste dans diverses études telles que [FON 88] [PRI 88].

### 5.22. Caractéristiques d'une telle exploration

*L'exploration* du domaine de la compilation d'architectures dédiées aux circuits de communication peut prendre la voie des études suivantes :

#### A. IDENTIFICATION DES CONTRAINTES DE DESCRIPTION (ou de langage d'entrée)

Si les caractéristiques générales des protocoles sont bien connues, le lien entre une description (par un langage fonctionnel, par un langage séquentiel, par des machines à états, etc...) d'un protocole donné et son comportement une fois implanté est encore un domaine nouvellement exploré, quand il s'agit d'implantation logicielle, et peu exploré, quand il s'agit d'implantation

matérielle.

Il s'agit de faire l'analyse des langages et des outils de description de protocoles (citons ESTELLE, LOTOŠ, ACT1), des langages de type VHDL ou des langages tels qu'OCCAM ou ADA, pour savoir si leur syntaxe et leurs *possibilités d'expression de la sémantique que nous visons* sont adaptés.

## B. IDENTIFICATION DES CONTRAINTES FONCTIONNELLES

Différentes caractéristiques des protocoles de communication conduisent à différents problèmes d'architecture à résoudre, au choix de différentes architectures à faire sous contraintes et à terme à bâtir dans le futur différents modèles pour ce faire. Citons des exemples de contraintes ou de caractéristiques :

- . liées à la reconnaissance des bornes des champs d'une trame : reconnaissance statique, reconnaissance dynamique (dont la connaissance est implicite, ou dont la connaissance est portée par le contenu d'un autre champ, en éclaté ou en continu);
- . liées à la reconnaissance du contenu d'un champ (immédiate, filtrée, ou différée à l'intérieur du circuit) et à l'exploitation immédiate ou différée de cette reconnaissance;
- . au traitement récursif de champ, au degré de parallélisme induit, etc...

## C. IDENTIFICATION DES CONTRAINTES LIEES AU PASSAGE DE L'ARCHITECTURE D'UNE MACHINE INTERMEDIAIRE A L'ARCHITECTURE D'UNE MACHINE PHYSIQUE

L'architecture qui sera générée par un compilateur ne sera que celle d'une machine intermédiaire. Le placage de l'architecture d'une machine intermédiaire sur une machine physique donnée ne peut faire abstraction des contraintes dites système. Par exemple :

- . les contraintes de communication et de synchronisation à l'interface,
- . les contraintes d'initialisation associées aux données devant se trouver sur le circuit,
- . l'existence des structures de données spécifiques (embarquées sur le circuit ou non).

### 5.3. CONCLUSION

Les deux esquisses de projet présentées sont caractéristiques de la démarche générale qui régit notre équipe. L'écho, si écho favorable il y a, que de tels projets ou projets-sœurs pourraient trouver dans le secteur de Recherche et Développement leur permettrait de démarrer et d'espérer aboutir à nouveau à des résultats utilisables par la communauté. Ainsi, nos efforts n'auraient-ils pas été fournis sans suite.



**BIBLIOGRAPHIE**



## REFERENCES

- [AGU 87] A Plausibility-Driven Approach to Computer Architecture Design. *U. Agüero, S. Dasgupta* . CACM, vol. 30, n° 11. 11/1987.
- [AMD 85] The AMD 7990 Family Ethernet Node. *AMD* . 1985.
- [AMS 85] La Réalisation d'une Bibliothèque de Cellules destinés aux Circuits Intégrés de Communication. *Convention CNET/CCI/AMS -ADR/LGI/PCCI* . 4/1985.
- [AMS 86] Système Hiérarchisé Utilisable par des Spécialistes Système. Projet d'Etude CNET CNS/CCI/AMS . 1986.
- [ANS 88] Architecture of a new ASIC data communication circuit intended for the ISO levels 3, 4, 5 : Network, Transport and Session. *Y. Ansade, Ng.X. Dang , M. Diaz-Nava , G. Michel, J. Rarivomanana , L. Sponga* . Actes du Congrès 1988 IEEE International Symposium on Circuits and Systems. Helsinki. 6/1988.
- [AOY 87] Role of VLSI in Telecom Age. *T. Aoyama* . Actes du Congrès 1987 Symposium on VLSI Circuits. Karuizawa. 5/1987.
- [APP 85] A Survey of System Programming Languages: Concept and Facilities. *W.P. Appelbe, K. Hansen* . Software Practice and Experience. Vol 15 (2). 2/1985.
- [APT 87.1] FACTOR, Réseau Local Industriel. *APTOR S.A* .1985.
- [APT 87.2] GASP. Logiciel de Synthèse de Systèmes Logiques, Combinatoires et Séquentiels. *APTOR S.A* . 1987.
- [APT 87.3] LUCKY-LOG. Simulateur logique. *APTOR S.A.* . 1987.
- [APT 87.4] Rapport phase 2 du projet MC3. Spécifications détaillées du composant MC3. Chapitre 1 : L'Interpréteur de Process et le Noyau Système. *APTOR S.A.* . Projet MC3. 29/6/1987.
- [APT 88] LUCAS. *APTOR S.A.* . 1988.
- [ARC 86] Spécifications Techniques d'Utilisation et de Raccordement ARCHITEL : Transport Session et Relations entre Couches. Notes Techniques NT/PAA/DIR/941 et NT/PAA/DIR/942. *ARCHITEL* . PAA/TIM/CNET PARIS A. Paris. 1986.

- [BAR 84] Conception d'un circuit Arbitre de Bus de Communication Multi-protocoles : ABCM. *Dante A. C. Barone* . Thèse de l'INPG. Grenoble. 11/1984.
- [BEL 85] Technology 85, Communications. *T.E. Bell* . IEEE Spectrum 1/1985.
- [BER 84] A Building Tool for Flexible Blocks Library. *J.M. Bergé, L.O. Donzelle, V. Olive, J. Rouillard, D. Rouquier* . Actes du Congrès ICCD. Port Chester. 10/1984.
- [BER 86] Development and Use of Flexible Block Libraries. *J.M. Bergé, L.O. Donzelle, V. Olive, J. Rouillard, D. Rouquier* . Actes du Congrès ESSIRC 86. Delft. 9/1986.
- [BEY 82] A Design Methodology based upon Symbolic Lay-Out and Integrated CAD Tools. *A.M. Beyls, B. Hennion, J. Lecourvoisier, G. Mazaré, A. Puissochet* . Actes du Congrès 19th DAC. 1982.
- [BOS 87] CAN Specification. Version 1.0. *R. Bosch, GMBH* . Stuttgart. 1987.
- [BOU 85] La Conception de Circuits Intégrés Silicium au CNET. *J. Boulvin, A. Girard, J.L. Lardy* . L'Echo des Recherches n° 121. 1985.
- [BOU 86] Lynx: an advanced deterministic CSMA-CD local area network prototype. *J. Boudenant, B. Feydel, A. Peyrache, P. Rolin* . Advanced Seminar on Real-Time Local Area Networks. INRIA. Bandol. 4/1986.
- [CAN 86] Conception de Circuits Intégrés MOS. *M. Cand, E. Demoulin, J.L. Lardy, P. Senn* . Collection Technique et Scientifique des Télécommunications. 1986.
- [CAN 88] Opérateur Paramétrable pour le Traitement du Signal. *M. Cand* . CNET CNS. 11/1988.
- [CAP 79] Generalized TDMA : the Multi-Accessing Tree Protocol. *J.I. Capetanakis* . IEEE Transactions on Communication. Vol 27. N° 10. 1979.
- [CAS 83] Conception Certifiée de Systèmes Distribués : un Exemple. *P. Caspi, N. Halbwachs* . ATP/CNRS. 10/1983.
- [CAS 84] CASSIOPEE, Note Technique CNET , NT/CNS/CCI/27, 5/1984
- [CCI 84] X25, niveau Paquet. Redbook. CCITT . 1984.
- [CCI 85] ISDN Protocols Specifications. Redbook. CCITT . 9/1985.

- [CHO 87] A CMOS VLSI Frammer Chip for a Broadband ISDN Local Access System. H.J. Chao, T.J. Robe, L.S. Smoot . Actes du Congrès 1987 Symposium on VLSI Circuits. Kuruizawa. 5/1987.
- [CIA 87] Livre Blanc sur les Capteurs Intelligents. *Commission Industrie-Administration pour la MEsure- AFCET* . CIAME . Paris. 1987.
- [CLA 78] An Introduction to Local Area Networks. *D. D. Clark, K.T. Pograd, D.P. Reed* . Proceedings IEEE Vol 66. 11/1978.
- [COR 81] Systèmes Informatiques Répartis. Concepts et Techniques. *Groupe CORNAFION* . Ed. Dunod. 1981.
- [DAN 76] Implémentation de la Station de Transport Version 2 du Réseau CYCLADES sur l'IBM 360/67 du CICG. *Ng.X. Dang*. Rapport de Contrat INPG - INRIA. 1976.
- [DAN 82] Local Area Networks for Distributed Process Control Systems. *Ng.X. Dang, G. Mazaré, G. Michel* . International Seminar on Synchronisation, Control and Communication in Distributed Computing Systems, PCL, London (UK), 20-24/9/1982.
- [DAN 83] Small Local Area Network with a Collision-Free Technique for connecting Heterogeneous Micro-Processors and Micro-Computers. *Ng.X. Dang, M. Diaz-Nava, N. Husovic* Microprocessing and Microprogramming, vol 12, 1983.
- [DAN 85.1] Circuits Intégrés VLSI dans les systèmes de communication. *Ng. X. Dang* . Séminaire au CNET CNS. Meylan. 4/1985.
- [DAN 85.2] Etude de Faisabilité de l'Intégration des Couches 3, 4, 5. *Ng.X. Dang, G. Michel* . Rapport Interne CNET CNS - IMAG/LGI. Grenoble. 7/1985.
- [DAN 85.3] Circuits Intégrés VLSI dans les systèmes de communication de données. *Ng.X. Dang, M. Diaz-Nava* . Actes du Congrès SupTélécom. Paris. 10/1985.
- [DAN 86.1] ILANS, Instrumentation Local Area Networks. *Ng.X. Dang, G. Michel* . Advanced Seminar on Real-Time Local Area Networks, ESRO, INRIA, Bandol, 4/1986.
- [DAN 86.2] Spécifications Fonctionnelles pour le Composant FIP Capteurs. *Ng.X. Dang, G. Michel* . Rapport de Synthèse. Groupe FIP Capteurs. 15/4/1986.  
Présentation au séminaire de la CIAME, groupe de travail "Capteurs Intelligents". Grenoble. 3/1988

[DAN 86.3] Rapport sur l'Analyse de la Complexité et de la Faisabilité du Contrôleur d'Accès CSMA/DCR. (200 p.). *Ng.X. Dang, L. Sponga* . Convention INRIA/SCORE - INPG/IMAG/LGI/PCCI. 6/1986.

[DAN 86.4] Design of a VLSI Communicating Circuit for an Industrial Local Network in Control Process and Automated Production. *Ng.X. Dang, M. Diaz-Nava, G. Michel* . Actes du Congrès EUROMICRO 86. Venise. 9/1986.

[DAN 87.1] Les circuits VLSI de communication de données : Architecture et Méthodologie. *Ng.X. Dang* . 9ième Journées Francophones sur l'Informatique. Liège. 1/1987.

[DAN 87.2] The effect of High Speed LANs on the design of the VLSI communicating circuits. *Ng.X. Dang, P. Rolin, L. Sponga, G. Votsis* . IFIP WG 6.4 Workshop on High Speed LANs. Aachen . 2/1987.

[DAN 88.1] Specific Data Structure intended for the Implementation of High Levels ISO standards : Associated and Dedicated Hardware. *Ng. X. Dang, C. Diot, I. Sabouni, L. Sponga*. Actes du Congrès EUROMICRO 88. Zurich. 8/1988.

[DAN 88.2] Aspects of Parallelism in the Architecture Definition of an ASIC Data Communication Intended for the ISO layers Network, Transport and Session. *Ng.X. Dang, C. Diot, I. Sabouni, L. Sponga, G. Michel, Y. Ansade, M. Diaz-Nava, J. Rarivomanana* . Actes du Congrès International Conference on Data Communication Technology. Limerick. 9/1988.

[DAN 88.3] A Direct Memory Access Module for Managing Linked List Structure : as generated by an Automatic Compiler of Flexible Blocks intended for VLSI communication circuits.. *Ng.X. Dang, I. Sabouni, L. Sponga* . Actes du Congrès International Conference on Microelectronics ICM 88. Alger. 11/1988.

[DAN 88.4] Rapports du projet CIVA2. Etude de Compilateurs de Silicium Spécialisés dans le domaine des Circuits Intégrés de Traitement des Protocoles de Communication.  
: phase 1 : Etude des Modules : DMA, FIFO, Hacheur, Reconnaissance de codes-fonctions (68 p.). *Ng.X. Dang, I. Sabouni, L. Sponga* . 7/1988.  
phase 2 : Etude des Opérateurs (66 p.). *Ng.X. Dang, P. Objois, I. Sabouni*.. 10/1988.

[DAV 80] Vectorizing and Priority Interrupts by means of PIA. *D.J. David* . EUROMICRO Journal, 6, 1980.

[DGA 88] Modèle de Référence des Réseaux Locaux Temps-Réel Militaires. Couche Transfert.GAM-T-103. DEI. *DGA* . Ministère de la Défense. Paris. 1988.

- [DIA 86.1] Proposition d'une Méthodologie de Conception de Circuits Intégrés de Communication. Réalisation d'un Communicateur pour le réseau local FIP. *M. Diaz-Nava* . Thèse de l'I.N.P.G. Grenoble. 7/1986.
- [DIA 86.2] Specification of a VLSI controller circuit for the FIP network. *M. Diaz-Nava, Ng.X. Dang* . International Symposium on Local Communication Systems, LAN and PABX, Toulouse, 26-28/11/1986
- [DIA 87] Estelle, une technique de description formelle des protocoles. *M. Diaz* . Actes des 9ième Journées Francophones sur l'Informatique. Liège. 1/1987.
- [DIO 88] Contribution à l'Etude de l'Implantation des Protocoles de Communication des Couches Hautes de l'ISO. Etude de performances : Maquette Fonctionnelle d'une Machine dédiée (MC3) versus TRANSPUTER. *C. Diot, Ng.X. Dang* . Rapport de Recherche. Laboratoire Génie Informatique. RR n° 741 I. 10/1988.
- [DOL 88] Bibliothèque de Cellules.  
Règles de synthèse pour les chemins de données (pour la connexion des parties opératives en tranches). *Dolphin S.A* . Marché d'études CIBLE. CNET CNS. 1988.
- [ESP 86] Communications Network for Manufacturing Applications. Phase 1, Implementation guide. *Esprit Project 955* . Bruxelles. 10/1986.
- [ESW 81] Collision-Free Access Control for Computer Communication Bus Networks. *K. P. Eswaran, V.C. Hamacher, G.S. Shedler* .  
. IBM Research Report RJ 3024. San Jose. California. 1981.  
. IEEE Transactions on Software Engineering. 11/1981.
- [FON 88] Synthèse automatique d'architectures. *A. Fonkua* . Thèse en cours de préparation. LGI/IMAG. 11/1988.
- [FLA 84] A Complete & Automatic System for Sequencer Design, *E. Flamand*, Actes du Congrès ICCD, New York, 1984.
- [FRA 81] The EXPRESSNET : a Local Area Network Integrating Voice and Data. *L. Fratta, F. Borgonovo, F.A. Tobagi* . Actes du Congrès International Conference on Performance of Data Communication Systems and Their Application. Paris. 9/1981.
- [FRA 83] An improved Access Protocol for Data Communication Bus Networks with Control Wire. *L. Fratta* . ACM 0-89791-089-3/1983.

- [GAL 84] Proposition d'un système de transmission série multiplexée pour les échanges d'informations entre des capteurs, des actionneurs et des automates réflexes, *D. Galara, J.P. Thomesse*, Livre Blanc, Ministère de l'Industrie et de la Recherche, 1984
- [GOL 82] Small area networks fit jobs too small for local nets. *A. Goldberg, C. Kaplinsky* . Electronics. 11/1982.
- [GOM 84] A Software Design Method for Real-Time Systems. *H. Goma* . CACM. 9/1984.
- [GUS 84] Computer Buses. A tutorial. *D.B. Gustavson* . IEEE Micro. 8/1984.
- [HAM 81] Collision-Free Local Area Bus Network Performance Analysis. *V.C. Hamacher, G.S. Shedler* .  
 . IBM Research Report RJ 3101. San Jose. California. 1981.  
 . Proceedings of 7th International Symposium on Computer Architecture. IEEE & ACM. La Baule. France. 1980.
- [HAR 85] Développement du RNIS en France : le projet RENAN. *D. Hardy* . Actes du Congrès "De Nouvelles Architectures pour les Communications", Paris, 10/1985
- [HAS 81] Opérateur Niveau Paquet X25 en FPLA. *M. Hascoet, J.C. Simon* . DEA TIDS. CCETT Rennes. 1981.
- [HAZ 84] Functional modeling for logic simulation. *H. El Hazem, G. Mazaré, M. Poize, A. Puissochet* . Actes du Congrès ICCD 84. Port Chester. 10/1984.
- [HEN 85] SHIVA, Système Hétérogène d'Implémentation et de Validation d'ARCHITEL. *A. Henriot* . AIS/T/17/1985. CCETT. 1985.
- [HER 87] Vehicle Interconnection Protocols. Layers 1 and 2. *P. Herbault* . PSA. Working Group 3 Structure. PROMETHEUS Eureka Project. 10/1987.
- [HOA 79] Communicating Sequential Process. *C.A.R. Hoare* . CACM. 4/1979.
- [IBM 86] IBM leaks performance details of forthcoming token ring network. *IBM* . Data Communications. 6/1986.
- [IEE 84] IEEE Local Area Network Standards. *IEEE Computer Standard Committee* . 1984.
- [IME 88] Scientific Report 1987. Division VLSI Systems and Design Methodologies *IMEC* . Louvain. 1987.

- [INM 87] Transputer T414 Reference Manual. *INMOS* . 1987.
- [INR 83] Bulletin de Liaison et de la Recherche en Informatique et Automatique. Les Réseaux Locaux. *INRIA* . n° 89. 1983.
- [INT 84.1] INA 960 Network Software. iSBC® 186/51 COMMputer™ Board. iSBC® 550 Ethernet Controller. 82586, 82588 LAN Controller. *INTEL*. 1984.
- [INT 84.2] iOSP™86, Operating System Software. 80130, Real-Time Operating Systems Processors. *INTEL* .1984.
- [ISD 85] ISDN Protocols Specifications. Redbook. CCITT. 9/1985.
- [ISO 84] Normes Réseau, Transport et Session. *ISO* .
- [JOL 83] Interconnexion par bus en environnement multiprocesseur. Principes de communication et techniques d'arbitrage. *Joly* . Thèse de Docteur Ingénieur. ENST. Paris. 6/1983.
- [JUL 87] A data-base interface for an integrated CAD system. *C. Jullien, A. Leblond, J. Lecourvoisier* . Actes du Congrès 23 rd DAC. Las Vegas. 1987.
- [KUN 88] Vehicle Distributed Executive VDX Proposal. PROMETHEUS Draft Specifications. *A. Kung, C. Serrano-Morales* . Renault S.A. Paris. 1988.
- [KAM 86] Protocols for Communications in the Factory. *M.A. Kaminsky Jr.* IEEE Spectrum. 4/1986.
- [KAN 86] Performance Measurements of a Data Link Protocol. *H. Kanakia, F. Tobagi* . Actes du Congrès ICC 86. Toronto. 6/1986.
- [LaR 88] Les nouveaux ordinateurs. Revue La Recherche. Octobre 1988.
- [LAR 86] Du rôle des bibliothèques dans la réduction des coûts et délais de conception pour les circuits intégrés spécifiques d'une application. *J.L. Lardy, J. Lecourvoisier, D. Rouquier* . Note Technique NT/CNS/CCI/58. 10/1986.
- [LAR 87] Les Circuits ASIC. *J.L. Lardy* . CNET CNS. 10/1987.
- [LeL 83] On Real-Time Distributed Computing. *G. Le Lann* . Actes du Congrès IFIP 83. Paris. 1983.

- [LeL 84] Protocole de gestion des accès multiples et réseaux locaux temps réel. *G. Le Lann* . Lettre d'Informations Réseaux, n° 13, 12/1984.
- [LEO 84] Factory Nets follow a MAP, *G. Leopold*. Electronics Week, 17/12/1984.
- [MAP 84] MAP. *National Computer Conference* . Las Vegas. 7/1984.
- [MAR 79] LISA : Communication Mechanism for Local Networks. *M. Marinescu* . Actes du Congrès First International Conference on Distributed Computing System. Huntsville. 10/1979.
- [MAR 87] Logiciels pour les couches Transport et Session ISO. *MARBEN Informatique* . 1987.
- [MC3 87.1] Projet MC3 : Définition et Validation de l'architecture de MC3 (VLSI pour protocoles de communication Session, Transport et Réseau). *APTOR S.A., DOLPHIN S.A., LGI/IMAG* .
- Rapport phase 1 : Les fonctionnalités à assurer par le composant MC3. (49 p. et annexes). *Ng. X. Dang, B. Lebayle, Y. Ansade* . 3/1987.
- [MC3 87.2] Rapport phase 2 du projet MC3 : Spécifications détaillées du composant MC3. Chapitre 2 : Les machines de transfert (55 p.). *Ng. X. Dang, L. Sponga* . 29/6/1987.
- [MC3 88.1] Rapport phase 3 du projet MC3 : Programme de description logico-fonctionnelle du circuit MC3 et Rapport d'évaluation sur les résultats de simulation. La machine de transfert (150 p.). *Ng. X. Dang, I. Sabouni, L. Sponga* . 31/3/1988.
- [MC3 88.2] Rapport phase 4 du projet MC3 : Maquette sur carte d'un simulateur du cœur de MC3. 4/1988.
- [MC3 88.3] Rapport phase 5 du projet MC3 : Mini-application de démonstration. 6/1988.
- [MEA 80] Introduction to VLSI Design. *C. Mead, L. Conway* . Addison-Wesley Ed. 1980.
- [MHS 87] Open Asic MacroCell I2C. I2C Protocol Controller. MHS Data Sheet. *MHS* . 10/1987.
- [MHS 88] Etude et Réalisation de Macro-Cellules pour Applications Spécifiques dans le domaine des Télécommunications. *MHS S.A.* . Marché d'études DGT. 1988.
- [MIC 82] CICS 81 : A VLSI based Local Area Network for Distributed Process Control. *G. Michel, J. Rouillard, G. Charles, D. Tranvaux* . Actes du Congrès 3rd International Conference on Distributed Computing Systems. Miami. 1982.



- [MOA 81] Spécification et Conception Sûre d'Automatismes Discrets et Complexes, basées sur l'utilisation du GRAFCET et des Réseaux de Pétri. *M. Moalla* . Thèse d'Etat. Université 1 de Grenoble. 7/1981.
- [MOD 82] MODWAY : Communication Utility Multi-Parallel Applications Industrial Standard. *GOULD-MODICON* . Internal Report. 1982.
- [MOK 84] Networks expand as PBXs get smarter, *N. Mokhoff* . Computer Design, 2/1984.
- [MOT 86] MC 68824. Preliminary Data Sheet. Token Bus Controller. *MOTOROLA* . 5/1986.
- [OBR 82] Etude Comparative de différentes Méthodes de Conception des Parties Contrôle des Microprocesseurs. *M. Obrebska* . Thèse de Docteur-Ingénieur de l'INPG. 6/1982.
- [OLI 82] Conception d'un Circuit Intégré Arbitre de Bus : ABC 90. *V. Olive, D. Rouquier* . Actes du Congrès AFCET. 10/1982.
- [OLI 83] A Systematic Method for the Synthesis of Control Part defined by GRAFCET. *V. Olive, D. Rouquier* . Actes du Congrès IFIP. Paris. 9/1983.
- [OLI 87] Programmation d'Automates sur Silicium. *V. Olive* . Thèse de l'INPG . 1987.
- [OST 87] ECOM 25 Controller for connecting to Public or Private X25 networks. *OST S.A.* 1987.
- [PIL 87] Simulation Logique de l'Interface Parallèle. *J.P. Pillou* . *DOLPHIN S.A* . Rapport phase 3 du projet MC3. 30/3/1988.
- [POU 82] The Cyclades Computer Network. *L. Pouzin & al* . Ed. North-Holland. ICCM Monography n° 2. 1982.
- [POW 81] Réseaux Locaux de Commande-Contrôle Sûrs de Fonctionnement. *D. Powell* . Thèse d'Etat. INP Toulouse. 1981.
- [PRI 88] Architectures pour le Codage d'Images. *G. Privat* . CNET CNS. 2/1988.
- [RAB 88] An evolutionary VLSI chip set for ISDN. *D.H. Rabaey, D. Sallaerts, R. Dierckx, M. Rahier*. Actes du Congrès 1988 IEEE International Symposium on Circuits and Systems. Helsinki. 6/1988.

- [RAR 88] Etude et Réalisation d'un Compilateur d'OCCAM pour la machine MC3. *J. Rarivomanana. APTOR S.A. . Notes personnelles. 1988.*
- [REN 82] Réalisation d'un système de communication par paquets à haut débit, destiné au réseau local en boucle du CCETT. *R. Renoulin, J.P. Lefranc, M.J. Takhedmit . Actes du Congrès AFCET, 1982.*
- [REI 83] TESS : Evalueur Topologique Prédicatif pour la Génération Automatique des Plans de Masse de Circuits VLSI. *R. A.D.L. Reis . Thèse de Docteur-Ingénieur de l'INPG. 1/1983.*
- [REY 84] IMOTHEP - Un Générateur Automatique d'Architecture pour Circuits Intégrés de Filtrage Numériques. *J.F. Reyssbrion . Thèse de l'INPG. 5/1985.*
- [ROD 86] FIP : A Bus for Industrial Instrumentation. *M. Rodriguez, J.P. Thomesse . Advanced Seminar on Real-Time Local Area Networks, ESRO, INRIA, Bandol, 4/1986.*
- [ROL 87] Mesure de Systèmes Répartis et Réseaux. *P. Rolin . Thèse d'Etat. Université de Rennes 1. 10/1987.*
- [ROU 86] La Compilation de Silicium. *D. Rouquier . Actes du 2ième Colloque National sur la Conception de Circuits à la Demande. Grenoble. 10/1986.*
- [ROU 87] Silicon Compilation Studies at CNS. *D. Rouquier . IFIP Workshop on Fast-Prototyping of VLSI. Ed. North-Holland. 1987.*
- [SCO 86] Analyse de la complexité et de la faisabilité du contrôleur d'accès CSMA-DCR. *Convention INRIA /SCORE-IMAG/LGI/PCCI . 6/1986.*
- [SHO 80] An Annotated Bibliography on Local Computer Networks. *J. Shock . XEROX Palo Alto Research Center. 1980.*
- [SUZ 81] Etude des Parties Opératives à Eléments Modulaires pour Processeurs Monolithiques. *A.A. Suzim . Thèse de Docteur-Ingénieur de l'INPG Grenoble. 11/1981.*
- [TAM 88] Design and Implementation of High-Speed Asynchronous Communication Ports for VLSI MultiComputer Nodes. *Y. Tamir, J.C. Cho . Actes du Congrès 1988 IEEE International Symposium on Circuits and Systems. Helsinki. 6/1988.*
- [TOB 86] Multi-Access Protocols and Real-Time Communications in LANs. *F. Tobagi . Advanced Seminar on Real-Time Local Area Networks. INRIA. Bandol. 4/1986.*

- [THO 87] TSVME 541. Carte Intelligente X25 à 2 canaux. *Thomson Composants Semi-conducteurs* . 1987.
- [TRANSPAC] X25.3. Spécifications techniques d'utilisation du réseau *TRANSPAC* .
- [TRE 82] VLSI Processors Architectures. *P.C. Treleaven* . IEEE Computer. 6/1982.
- [UTE 87] Bus FIP pour échange d'information entre transmetteurs, actionneurs et automates. Couche Liaison de données. C46-603. 46/GE6-117. *Union Technique de l'Electricité* . Paris. 1987.
- [VOT 86] Spécifications Système d'un circuit VLSI contrôleur d'accès pour les réseaux Locaux Temps-Réel. *G. Votsis* . Thèse de l'Université de Paris 11. 12/1986.
- [WD 84] 2501, HDLC Controller. *Western Digital* . 1984.  
WD 2511A X25 Packet Network Interface (LAP B). *Western Digital* . 1987.
- [ZIM 84] On Protocol Engineering. *H. Zimmermann* . Rapport CNET Paris A. Paris. 1984.

**AUTORISATION de SOUTENANCE**

---

VU les dispositions de l'article 5 de l'arrêté du 16 avril 1974

VU les rapports de Messieurs

- . R. GERBER , Professeur
- . D. ETIEMBLE , Professeur
- . G. MAZARE , Professeur

Monsieur **NGUYEN-XUAN-DANG Michel**

est autorisé à présenter une thèse en soutenance pour l'obtention du grade de  
**DOCTEUR D'ETAT ES-SCIENCES.**

Fait à Grenoble, le 6 décembre 1988

12 12 1988

Le Président  
de l'Université Joseph Fourier  
P.o. Le Président,  
Le Vice-Président Recherche

  
A. NEMOZ

Pour le Président de l'I.N.P.-G.  
et par délégation,  
le Vice-Président  
P. VENNEREAU









## RESUME DE LA THESE

Ce rapport traite de l'intégration de protocoles de communication de données dans des circuits intégrés. Cette intégration est examinée sous deux aspects : architecture fonctionnelle et méthodologie de conception.

Deux classes de circuits de communication ont été étudiées : la première traite des méthodes d'accès série ou parallèles et la deuxième concerne l'implantation des couches hautes du modèle OSI, allant de la couche LLC d'IEEE à la couche Session.

L'approche architecturale est propre aux architectures multiprocesseurs. Cependant, les deux aspects internes de ce domaine cohabitent : l'aspect multi-unités asynchrones câblées pour la première classe donnée de circuits de communication et l'aspect processeur dédié à usage multiple (à architecture parallèle) de traitement de communication pour l'autre classe.

La méthodologie de conception des circuits de communication est basée sur la définition et le développement des bibliothèques d'opérateurs flexibles spécialisés et des bibliothèques de modules paramétrables spécialisés. Nos expériences menées dans le cadre de la conception du circuit FIP-VLSI et de la validation de l'architecture du circuit MC3 sont décrites.

Un des buts à terme est de produire des compilateurs de silicium de plus en plus performants pour préparer les études sur la synthèse automatique des circuits intégrés, dédiés dans notre cas à la communication.