

EX-CRN-PN 82-05



14 MARS 1983

C₁
C.R.N.

erches nucléaires de Strasbourg

CRN/PN 82-05

THESE

présentée

pour obtenir le grade de

DOCTEUR DE 3^{ème} CYCLE

Spécialité: Electronique et Instrumentation

par

Philippe KLEIN

**APPORT DES PROCESSEURS MICROPROGRAMMABLES
AU SYSTEME D'ACQUISITION ET AU CONTROLE
EN LIGNE DE L'EXPERIENCE NA10**

CERN LIBRARIES, GENEVA



CM-P00050235



Institut National
de Physique Nucléaire
et de Physique
des Particules

Université
Louis Pasteur
de Strasbourg

Thesis-1982-Klein

THESE

CRN/HE 82-05

presentee

A L'UNIVERSITE LOUIS PASTEUR DE STRASBOURG

pour obtenir le grade de

DOCTEUR DE SPECIALITE

Electronique et Instrumentation

par

Philippe KLEIN

**APPORT DES PROCESSEURS MICROPROGRAMMABLES
AU SYSTEME D'ACQUISITION ET AU CONTROLE
EN LIGNE DE L'EXPERIENCE NA10**

Soutenue le 16 Novembre 1982 devant la Commission d'Examen:

MM. G. WEILL

J.M. MEYER

G. SUTTER

J.J. BLAISING

J. LECOQ

President

Examineurs

LISTE DES PROFESSEURS, MAITRES DE CONFÉRENCES
DIRECTEURS ET MAITRES DE RECHERCHE CNRS ET INSERM

Président	Professeur	F.MARCOUX
Vice-Présidents	Professeur	Ph.ROPARTZ
	Professeur	H.DURANTON
Présidents honoraires	Professeurs	G.OURISSON - P.KARLI
Secrétaire Général	Monsieur	G.KIEHL

U.E.R. DES SCIENCES MÉDICALES

U.E.R. des Sciences Médicales	: Directeur	Marc DORNER
U.E.R. des Sciences Biomédicales	: Directeur	André KIRN

Doyens honoraires : J.CALLOT - J.CLAVERT - Fr.ISCH.

Professeurs honoraires : A.BASSET - J.BENOIT - J.CALLOT - J.CLAVERT - E.FORSTER - G.GREINER - Ch.GROS - A.JUNG - P.MANDEL - H.METZGER - P.MULLER - A.ROHMER - F.SCHMID - E.SCHNEEGANS - J.SEROR - J.STAHL - J.VEDRINE - P.VINTEMBERGER - J.WARTER - G.WINCKLER.

Professeurs :

M.ADLOFF N.APROSIO Cl.ARON L.ASCH A.BATZENSCHLAGER P.BEYER P.BLOCH R.BLOCH R.BOCKEL Cl.BOLLACK A.BRINI A.BRONNER F.BUCHHEIT P.BUCK G.BURGHARD P.CHAMBON J.CHAMBRON M.CHAMPY A.CHAUMONT M.COLLARD C.CONRAUX P.DELLENBACH M.DORNER R.EBTINGER M.FABRE L.FINCKER R.GANDAR P.GAUTHIER-LAFAYE	Chirurgie générale Anatomie et Organogénèse Histologie Rhumatologie Anatomie Pathologique Clin. de Pédiatrie et Puericulture Radiologie Pharmacologie Hépatologie, Gastro-entérologie Urologie Ophtalmologie Clin. Ophtalmologique Neurochirurgie Clin. Chirurgicale des Enfants Clin. et Prophylaxie de la Tuberc. Biochimie Physiologie biologique Stomatologie Médecine légale et Méd.Sociale Neurologie Oto-Rhino-Laryngologie Gynécol. et Obstétrique Clin. Médicale B Psychiatrie infantile Histologie Clin. Médicale B Clin. Gynécol. et Obstétricale Anesthésiologie	J.GRENIER E.GROSSHANS P. HABEREY J.HERAN L.HOLLENDER J.L.IMPES Fr.ISCH L.ISRAEL H.JAHN J.JUIF Th.KAMMERER P.KARLI B.KELLER F.KEMPF I.KEMPF R.KIENY A.KIRN J.G.KORITKE M.KREMER D.KLOTZ G.LANG J.LAVILLAUREIX Y.LEGAL J.M.LEVY J.M.MANTZ F.MARCOUX J.MARESCAUX Ch.MARX S.MAYER	Chirurgie générale Clif. Dermatol. et Syphilitigraphique Physiologie Méd. et Chir. Expérim. et Comparée Chir. Digestive et Générale Pharmacologie Rééducation fonctionnelle Psychiatrie d'Adultes Néphrologie Pédiatrie et Puericulture Clin. Psychiatrique Neurophysiologie Clin. Gynécol. et Obstétricale Radiologie Orthopédie et Traumatologie Pathol. et Clin. Sémio. Chir. Virologie Anatomie Normale Parasitol. et Pathol. tropicale Neurologie Orthop. et Traumatologie Hygiène Anatomie pathologique Pédiatrie, Génétique médicale Réanimation Médicale Médecine du Travail Histologie Physiologie Hématologie	J.MEHL G.METHLIN B.METZ R.MINCK G.MORAND E.PHILIPPE (dét.) R.RENAUD P.REVILLE Ph.REYS E.ROEGEL F.ROHMER J.V.RUCH Y.RUMPLER A.SACREZ G.SCHAFF E.SCHVINGT J.SCHWARTZ A.SIBILLY L.SINGER F.STEPHAN D.STORCK G.VINCENDON R.VOEGTLIN A.WACKENHEIM P.WARTER J.P.WEILL D.WILLARD J.P.WITZ	Médecine du Travail Physiologie biologique Physiologie appliquée Bactériol. Viro. Immunol. gén. Chirurgie thoracique Anatomie pathologique Gynécologie et Obstétrique Endocrinol. Métabol. et Nutrit. Chir. Gén.-Hôp. L. Pasteur Colmar Pneumologie Phtisiologie Clin. Neurologique Biologie médicale Embryol. et Morphol. expérim. Cardiologie Physiologie Clin. Chir. Orthop. et Traum. Adult. Pharmacol. et Méd. Expérim. Clinique Chirurgicale A Psychiatrie d'Adultes Pathol. Gén. et Expérimentale Clinique médicale A Biochimie Thérapeutique Radiologie Radiologie Gastro-Entérol. et Hydrol. Thérap. Pédiatrie, Génétique Médicale Chirurgie thoracique
---	---	---	---	---	---

Professeur associé : G.VAN ALPHEN - Ophtalmologie.

Professeur conventionné : E.WEIL - Toxicologie industrielle.

Maitres de Conférences Agrégés :

S.BABIN P.BAREISS P.BOURJAT C.BRECHENMACHER J.M.BROGARD A.CONSTANTINESCO J.P.DUPEYRON B.EISENMANN J.FLAMMENT J.GEISERT J.P.GERHARD P.GERLINGER E.HEID M.IMLER D.JAECK A.JAEGER M.JAESEL	Orthopédie et Traumatologie Médecine interne Radiologie Cardiologie Médecine interne Physique biologique Anesthésiologie Chirurgie cardio-vasculaire Ophtalmologie Pédiatrie, Génétique médicale Ophtalmologie Embryologie Dermato-vénérologie Médecine interne Chirurgie générale Réanimation médicale Rééducation fonctionnelle	P.KEHR R.KEILING J.KEMPF T.KIEN TRUONG G.KLOTZ F.KUNTZMANN J.M.LANG D.MAITROT J.L.MANDEL J.MARESCAUX J.MARK J.MESSER C.MEYER H.MONTEIL F.OBERLING G.OBERT	Orthopédie et Traumatologie Cancérologie Biochimie Parasitologie Oto-Rhino-Laryngologie Médecine interne Maladies du sang Neurochirurgie Biochimie Chirurgie générale Biochimie Pédiatrie, Génétique médicale Chirurgie générale Bactériologie Hématol. Maladies du sang Bactériologie Virologie	J.C.OTTENI G.PAULI J.RITTER M.ROOS P.SAUVAGE G.SAVA J.P.SCHIEBER G.SCHLAEDER J.L.SCHLIENGER H.SICK C.STOLL J.D.TEMPE J.TONGIO J.P.WALTER J.M.WARTER A.WILK	Anesthésiologie Pneumologie Phtisiologie Gynécol. et Obstétrique Histologie Chirurgie infantile Chirurgie générale Physiologie Gynécologie et Obstétrique Médecine interne Anatomie et Organogénèse Pédiatrie, Génétique médicale Réanimation médicale Radiologie Radiologie Neurologie Stomatologie
---	---	--	---	---	---

Directeur de Recherche : A.PETROVIC* - Physiologie - M.JACOB* - Biochimie

Maitres de Recherche :

M.GAUCHERIE+ G.GOMBOS+ K.HAFFEN-STENGER+ G.LECLERC+	Thermologie biomédicale Neurochimie Endocrinologie Chimie organique	G.REBEL+ R.RECHENMANN+ M.SENSENBRENNER+ J.VELLY+	Neurochimie Biophys. des Rayonnements Neurochimie Pharmacologie	N.VIRMAUX-COLIN+ J.J.VOGT+ A.WAKSMANN+	Neurochimie Thermophysiology Neurochimie
--	--	---	--	--	--

U.E.R. D'ODONTOLOGIE

Directeur **Robert FRANK**

Professeurs :

M.DOCQ R.FRANK	Dentisterie opératoire Biologie et Matière fondamentale	P.KLEWANSKY J.LITZLER	Parodontologie Prothèse
-------------------	--	--------------------------	----------------------------

Professeurs de premier grade :

E.ALLEMANN M.BASTIAN C.BOLENDER A.COMTE	Dentisterie opératoire Prothèse Orthopédie dento-faciale Dentisterie opératoire	R.HAAG J.L.LACOSTE M.LANGER M.LEIZE	Pathol. et Therap. dentaires Orthopédie dento-faciale Prothèse Prothèse	P.NICOLAS J.J.ROTH A.SCHLIENGER J.SOMMERMATER	Pathol. et Therap. dentaires Parodontologie Prothèse Parodontologie
--	--	--	--	--	--

Professeurs de deuxième grade :

W.BACON P.CAHEN J.P.CHARLIER	Orthopédie dento-faciale Biologie et Matière fondamentale Orthopédie dento-faciale	B.KAESS H.TENENBAUM	Pathol. et Therap. dentaires Parodontologie
------------------------------------	--	------------------------	--

U.E.R. DES SCIENCES PHARMACEUTIQUES

Directeur **Pierre MÉTAIS**

Doyens honoraires : P.DUQUENOIS — N.HASSELMANN — G.DIRHEIMER.

Professeurs honoraires : P.CORDIER — J.P.EBEL — G.GAZET du CHATELIER — P.JAEGER.

Professeurs :

R.ANTON R.CARBIENER G.DIRHEIMER G.FERARD A.GAIRARD D.GERARD M.HASSELMANN C.HASSELMANN (dét.) L.JUNG	Pharmacognosie Botanique Toxicologie Chimie biologique Physiologie Physique et Biophysique Chim. Anal. et Bromatol. Chimie Analytique Pharmacie chimique	J.C.KOFFEL H.LAMI Y.LANDRY C.LAPP P.LAUGEL G.LAUSTRIAT A.LUGNIER J.MALGRAS C.MATHIS P.METAIS	Pharmacie chimique Mathématiques Pharmacologie Chimie générale et minérale Chimie analytique Physique Toxicologie Immunologie Pharmacie galénique Biochimie	B.PESSON Ph.POINDRON J.SCHREIBER A.STAHL A.STAMM J.C.STOCLET D.VIDON C.G.WERMUTH	Parasitologie Virologie Chimie organique Biochimie pharmaceutique Pharmacie galénique Pharmacodynamie Bactériologie Chimie organique
---	--	---	--	---	---

Professeur associé : A.WEINREB - Pharm. biophys.

Professeur conventionné : B.ROTH-SCHECHTER - Pharmacodynamie

Maitre de Recherche : I.N.S.E.R.M. : J.BIETH - Enzymologie

U.E.R. DE SCIENCES HUMAINES

U.E.R. de Géographie

Directeur **Pierre MICHEL**

U.E.R. des Sciences du Comportement et de l'Environnement

Directeur **Bruno WILL**

Professeurs honoraires : E.JUILLARD — R.RAYNAL.

Professeurs :

J.M.AVENARD L.LEGRAND M.LIEURY P.LIMOUZIN	Géographie Sciences de l'Education Psychol. expérim. Géographie	P.MICHEL A.MOLES H.NONN H.REYMOND	Géographie Psychologie sociale Géographie Géographie	R.SCHWAB A.TABOURET-KELLER M.TARDY J.TRICART	Géographie Psychologie Psycho-pédagogie Géographie
--	--	--	---	---	---

Directeur de Recherche : C.N.R.S. : S.RIMBERT - Géographie

U.E.R. DES SCIENCES ÉCONOMIQUES

Directeur **X...**

Doyens honoraires : P.CHAMLEY — J.P.FITOUSSI — R.DOS SANTOS FERREIRA.

Professeurs :

Ph.ARTZNER F.BILGER A.CHABERT	Mathématiques Sciences Economiques Sciences Economiques	P.CHAMLEY R.DOS SANTOS FERREIRA J.P.FITOUSSI (dét.)	Sciences Economiques Sciences Economiques Sciences Economiques	LAGERARD-VARET G.KOENIG J.J.OBRECHT	Sciences Economiques Sciences Economiques Gestion
-------------------------------------	---	---	--	---	---

Chargés de Conférences : R.ERBES — A.LOSSER

U.E.R. DES SCIENCES EXACTES

Mathématiques
Sciences Physiques et Chimiques
Sciences de la Matière
Sciences de la Vie et de la Terre
Sciences du Comportement et de l'Environnement
Ecole d'Application des Hauts Polymères
Ecole Nationale Supérieure de Chimie
Observatoire
Physique du Globe
Ecole Nationale Supérieure de Physique

Directeur **Gérad SCHIFFMANN**
 Directeur **Henri BENOIT**
 Directeur **Jean José FRIED**
 Directeur **Geneviève LEBEURIER**
 Directeur **Bruno WILL**
 Directeur **Morand LAMBLA**
 Directeur **Marc DAIRE**
 Directeur **Alphonse FLORSCH**
 Directeur **Roland SCHLICH**
 Directeur **Gilbert SUTTER**

Doyens honoraires : P.LACROUTE - J.H.VIVIAN - G.MILLOT.

Professeurs honoraires : J.BRENET - Mme M.BRINI - J.BYE - H.CARTAN - C.CHABAUTY - A.CHRETIEN - J.DENY - Mile A.GAGNIEU - Mile S.GILLET - S.GORODETZKY - R.HOCART - P.JOLY - P.LACROUTE - R.LECOLAZET - G.LEMEE - P.LHERITIER - A.LICHTNEROWICZ - A.MAILLARD - G.MILLOT - L.NEEL - J.PARROD - A.ROCHF - R.ROHMFR - J.P.ROTHE - L.SACKMANN - Ch.SADRON - H.SAUCIER - F.STUTINSKY - H.VILLAT - J.H.VIVIAN - Et.WOLFF.

Maître de conférence honoraire : R.WEIL.

Professeurs :

J.P.ADLOFF Chimie nucléaire
 R.ARMBRUSTER Physique
 P.ARTZNER Mathématiques
 V.AVANISSIAN Analyse supérieure
 G.BARBANÇON Mathématiques
 F.BECKER Physique mathématique
 N.BEFORT Biochimie
 G.BELLIARD Botanique
 C.BENEZRA Dermato-Chimie
 H.BENOIT Physicochim.macromol.
 P.BENVENISTE Physiologie végétale
 D.BERNARD Méth.math.de la physique
 J.C.BERNIER Chimie générale
 J.BONNIN Géophysique interne
 Y.BOULLANGER Biochimie
 J.F.BOUTOT Mathématiques
 J.BROSSAS Chimie macromoléculaire
 R.BROUILLARD Chimie org.et Chimie analyt.
 C.BURGGRAF Minéralogie
 H.BURNAGE Mécanique des Fluides
 M.CARA Géophysique interne
 R.CERF Physique générale
 P.CHARTIER Chimie
 P.CHEVALLIER Physique
 A.CLAUSS Chimie
 A.COCHÉ Physique nucléaire
 M.DAIRE Chim.phys.indus.et Sc.des Matér.
 H.DANAN Phys.Atom.et Phys.du Solide
 E.DANIEL Physique expérimentale
 M.DAUNE Biophysique
 J.DEHAND Chimie générale
 A.DELUZARCHE Chimie
 J.DEMUYNCK Chimie organique
 B.ERESMANN Informat.fondam.et appl. (IUT)
 G.DUNOYER de Géologie
 SEGONZAC
 H.DURANTON Botanique
 J.P.EBEL Biochimie
 J.P.EBERHART Minéralogie
 B.ERESMANN Biochimie
 V.ERN Physique
 J.FARAUT Mathématiques
 P.FEDERLIN Chimie
 P.FELTZ Physiologie animale

X.FERRIQUE Mathématiques
 M.FINK Acoustique moléculaire
 J.G.FISCHER Chimie
 D.FOATA Mathématiques
 E.FOLLENIUS Zoologie
 J.J.FRIED Mécanique des Fluides
 D.FROELICH Chim.gén.Chim.physique
 A.FUCHS Mécanique rationnelle
 J.C.GALL Géologie
 A.GALLMANN Physique
 F.GAUTIER Physique
 R.GERARD Mathématiques
 G.GLAESER Mathématiques
 Cl.GODBILLON Mathématiques
 A.GOLTZENE Optique, phys.atom.et mol., phys.du sol et cristallographie
 M.GOUNOT Botanique
 M.GROSSMANN Physique
 M.GROSS Chimie physique
 Y.GUEGUEN Géophysique interne
 L.HIRTH Microbiologie
 C.JASCHEK Astronomie
 G.JONARD Virologie
 J.P.JOUANOLOU Mathématiques
 F.JUNDT Phys.nucl.et corp.et théor.phys.
 T.JUTEAU Minéralogie
 C.KEDINGER Chimie
 A.KIENEMANN Chimie appl.et Génie chim.
 R.KIRSCH Zoologie
 F.LACROUTE Biologie végétale
 J.C.LAFON Informatique appliquée
 A.M.LAMBERT Biologie
 M.LAMBLA Chimie générale
 G.LEBEURIER Microbiologie
 J.L.LEIBENGUTH Chim.appl.et Chim.des matér.
 J.LEITE-LOPES Phys.nucl.et corpusculaire
 P.LEMOINE Chimie
 M.LEROY Chimie
 J.C.LIONS Zoologie et Ecologie
 F.LLOOR Immunologie
 J.LUCAS Géologie
 D.MAGNAC Physique
 J.MARTINET Mathématiques
 P.MIALHE Physiologie animale
 A.MICHARD Géologie

M.MIGNOTTE Informatique
 P.MIRABEL Chimie
 G.MONSONEGO Physique théorique
 B.MORIN Mathématiques I
 P.NANOPOULOS Probabilités et statistiques
 J.OSBORN Chimie
 E.OSTERTAG Electr. et Electrotechnique
 G.OURISSON (dét.) Chimie
 J.M.PAULUS Chimie générale
 J.P.RAMIS Mathématiques générales
 G.REEB Topologie
 Ph.RICHARD Physiologie animale
 J.J.RIEHL Chimie
 P.RIMMELIN Chimie (IUT)
 Cl.ROBERT Physique
 Ph.ROPARTZ Psycho-Physiologie
 J.ROUX Botanique
 J.P.SAUVAGE Chimie théor.et Chim.phys.(IUT)
 F.SCHALLER Biologie générale
 G.SCHIFFMANN Mathématiques
 A.SCHMITT Physique
 P.SCHMITT Psycho-physiol.et physiol.du comp.
 J.P.SCHWING Chimie
 M.J.SCHWING Chimie physique
 J.C.SENS Phys.nucl.et corp.et théorie phys.
 M.SIESKIND Physique
 G.SOLLADIE Chimie organique
 J.SOMMER Chimie appliquée
 G.SUTTER Physique électronique
 Ch.TANIELIAN Chimie (IUT)
 Y.TARDY Géologie
 J.TERRISSE Chimie
 J.J.THIEBOLD Biologie animale
 D.VIAUD Mathématiques
 R.VOLTZ Physique théorique
 J.H.WEIL Biochimie
 G.WEILL Physique
 R.WEISS Chimie
 P.L.WENDEL Physique
 B.WILL Psycho-Physiologie
 C.WINTER Chimie
 C.WIPPLER Physicoch.des Hts Polymères
 J.WUCHER Physique
 B.WURTZ Biochimie

Professeur émérite : G.MILLOT - Géologie Paléontologie.

Professeur adjoint : J.SITTLER - Géologie

P.ROUVEROT - Physiologie respiratoire P.DEJOURS Physiologie respiratoire.

Professeurs Associés :

M.R.CHURCHILL Chimie
 B.J.COLE Physique
 M.E.CONSTANTIN Chimie

E.CROUSE Biochimie
 H.DUGAS Chimie
 C.PIERPONT Chimie

A.SIDOANS Géologie
 R.WEISS Chimie

Professeurs conventionnés : P.BEY - Chimie - P. BOUVEROT - Physiol.respirat. - P.DEJOURS - Physiol.respirat. - F.FARKAS - Phys. - Y.NAKATANI - Chimie.

Astronomes adjoints : A.FLORSCH - A.FRESNEAU

Astronome adjoint associé : M.JASCHEK - Astronomie

Physiciens adjoints (Physique du Globe) : P.HOANG TRONG - Géophysique int. R.MONTIGNY - Géophysique int. E.PETERSCHMITT - Géophysique int.

Directeurs de Recherche C.N.R.S. :

P.ALBRECHT Chimie
 J.F.BIELLMANN Chimie
 P.BOUVEROT Physiologie respiratoire
 P.DEJOURS Physiologie respiratoire
 M.FRANCK-NEUMANN Chimie organique
 A.KNIPPER Phys.nucl.et corpusculaire

A.KOVACS Physicoch.macromoléculaire
 J.MARCHAL Physicoch.macromoléculaire
 P.A.MEYER Mathématiques
 A.J.P.MEYER Physique
 A.PORTE Biologie cellulaire
 P.REMPF Physicoch.macromoléculaire
 R.SCHLICH Géophysique marine

A.SKOUlios Physicoch.macromoléculaire
 M.VAN REGENMORTEL Virologie
 R.VAROQUI Physicoch.macromoléculaire
 A.VEILLARD Chimie moléculaire
 R.ZANA Physicoch.macromoléculaire
 A.ZUKER Physique théorique

Maitres de Recherche C.N.R.S. :

J.Ch.ABBE Physicoch.ds interactions et ds interfaces
 E.ASLANIDES Phys.nucl.et corpusculaire
 F.BECK Phys.nucl.et corpusculaire
 G.BECK Biochimie
 J.P.BECK Physiologie
 R.BERTINI Physique nucléaire
 M.BONHOMME Géologie
 H.BRAUN Physique nucléaire
 P.BRAUNSTEIN Chimie
 M.C.CADEVILLE Physique des Solides
 H.CALLOT Chimie
 S.CANDAU Physique
 M.CHAMPAGNE Biophysique
 J.CHEVALLIER Physique nucléaire
 J.P.COFFIN Phys.nucl.et corpusculaire
 A.CORET Physique
 M.CROISSIAUX Phys.nucl.et corpusculaire
 D.DISDIER Phys.nucl.et corpusculaire
 J.DOUBINGER Géologie
 F.DURST Physiologie végétale
 S.EL KOMOSS Physique
 E.FRANÇOIS Physicoch.macromoléculaire
 E.FRANTA Physicochimie moléculaire
 J.M.FRIEDT Physicoch.ds interactions et ds interfaces
 B.FRITIG Virologie
 Y.GALLOT Physicoch.macromoléculaire
 J.P.GERBER Phys.nucl.et corpusculaire
 R.GIEGE Biochimie
 Ph.GRAMAIN Physicoch.macromoléculaire
 J.B.GRUN Physique

J.HERZ Physicoch.macromoléculaire
 J.HOFFMANN Biologie animale
 G.JENNER Chimie
 G.KAUFMANN Chimie
 G.KEITH Biochimie
 J.P.KINTZINGER Chimie
 B.KOCH Physiologie
 E.KOCHANSKI Struct.et dynam.molécul.
 B.LANG chimie de coordination
 J.LANG Cristallographie
 P.LAURENT Physicoch.macromoléculaire
 A.LEJEUNE Physiol.comparée des régul.
 Cl.ERAY Phys.nucléaire théorique
 F.LEYENDECKER Physiol.comparée des régul.
 B.LOTZ Chimie
 B.LUU Physicoch.macromoléculaire
 G.MAIRE Chimie organique
 A.MALAN Chimie
 E.MARCHAL Physiologie respiratoire
 R.MORAND Physicoch.mol.et macromol.
 Th.MULLER Phys.nucl.et corpusculaire
 G.MUNSCHY Chimie
 M.NAUCIEL-BLOCH Phys.nucl.et corpusculaire
 A.NICOLAIEFF Chimie
 H.PAQUET Physique
 M.PATY Physique des Solides
 Cl.PICOI Virologie végétale
 L.PINCK Géologie
 Phys.nucl.et corpusculaire
 Physicoch.macromoléculaire
 Biologie cellulaire

P.POIX Chimie
 J.POUYET Biophysique
 B.REES Chimie
 P.REMY Biochimie
 J.RINGEISSEN Physique
 J.P.ROTH (s.Dir. Inst.de rech.) Physicoch.macromoléculaire
 R.SCHANTZ Physiologie végétale
 F.SCHEIBLING Phys.nucl.et corpusculaire
 F.SCHUBER Chimie organique
 N.SCHULZ Phys.nucl.et corpusculaire
 C.SCHWAB Physique
 R.SELTZ Phys.nucl.et corpusculaire
 P.SIFFERT Phys.nucl.et corpusculaire
 Cl.SITTLER Géologie
 M.E.STOECKEL Biol.des interactions cellulaires
 Cl.STRAZIELLE Physicoch.macromoléculaire
 M.SUFFERT Phys.nucl.et corpusculaire
 J.C.THIEFFRY Chimie
 K.TRAORE Physicoch.atom.et ionique
 P.WAGNER Phys.nucl.et corpusculaire
 G.WALTER Phys.nucl.et corpusculaire
 Fr.WEBER Géologie
 J.P.WENIGER Zoologie
 J.WITZ Biologie cellulaire
 R.WOLFF Chimie
 G.P.ZIELINGER Physique
 L.ZIL LIOX Mécanique des Fluides

A MA FEMME
A MES PARENTS
A MES SOEURS ET FRERE
A MES AMIS

P L A N

	Page
INTRODUCTION	2
CHAPITRE I	
I. CARACTERISTIQUES DE L'EXPERIENCE NA10	3
1. L'appareillage	
1.1. La cible	
1.2. L'absorbeur	
1.3. Les hodoscopes R1 à R4	
1.4. L'aimant	5
1.5. Les hodoscopes P1 et P2	
1.6. Les chambres MWPC	
2. Le système de déclenchement	6
2.1. Principe	
2.2. La logique associée	8
2.2.1. La coïncidence V	
2.2.2. La matrice de coïncidence	
2.2.3. La logique intersextants	9
3. Le système de mémorisation	10
4. Structure de l'information à acquérir	
II. LE SYSTEME D'ACQUISITION	13
1. Les fonctions d'un système d'acquisition	
2. Le système d'acquisition "classique" de l'expérience NA10	15
2.1. Architecture	
2.2. Caractéristiques du transfert CAMAC	
2.3. L'ensemble informatique attaché à l'expérience	16
2.3.1. Le matériel	
2.3.2. Le logiciel	
a) Le programme d'acquisition	
b) Le programme de traitement	

3. Limites du système d'acquisition classique NA10	19
3.1. Limites liées aux caractéristiques matérielles	
3.1.1. Manque de sélectivité du déclenchement	
3.1.2. Le temps de réponse RMH-NORD	22
3.2. Limites liées à la structure informatique de l'expérience	
3.2.1. Le traitement des données par le système NORD	
3.2.2. Le contrôle de l'expérience	24
3.3. Conclusion	
III. LE SYSTEME DE FILTRAGE	25
1. Architecture	
1.1. Réduction du flux de données	
1.1.1. Déclenchement logiciel	
1.1.2. Filtrage en ligne	
1.2. Masquage des temps morts	26
1.3. Choix du processeur de filtrage	27
2. Paramètres du filtrage	29
2.1. Détermination du traitement	
2.2. Précision du calcul de masse	
2.3. Vitesse de traitement	
3. Choix des opérateurs câblés	30
3.1. Recherche du plus grand moment transverse	
3.2. Multiplieur	
4. Le système d'acquisition avec EVB	35
4.1. Architecture	
4.2. Acquisition d'un événement	
4.3. La synchronisation GESPRO-système d'acquisition	39
4.3.1. L'interruption SDI	
4.3.2. Les registres d'état et de contrôle	40
4.3.3. Prise en compte du signal d'acquisition	
5. Le calculateur GESPRO	44
5.1. Architecture	
5.1.1. L'unité centrale	
5.1.2. Les périphériques	46
a) Module d'interruptions	
b) Opérateur de décalage	

c) Le multiplieur	
d) La mémoire centrale	48
e) Les cartes de recherches du plus grand P_T	
5.2. Format des microinstructions	
5.3. Les instructions	51
5.3.1. Format	
5.3.2. Adressage	
5.3.3. Déroulement d'une instruction	
5.4. Le jeu d'instructions GESPRO	55
CHAPITRE II	
I. METHODOLOGIE DE PROGRAMMATION DU SYSTEME DE FILTRAGE	56
1. Les traitements liés aux signaux de contrôle	
2. Les traitements de sélection	
II. LE PROGRAMME DE FILTRAGE	60
1. Les tâches GESPRO	
1.1. Les tests de sélection	
1.2. Les tests de contrôle	61
1.3. L'histogrammation	62
2. Organigramme du programme de filtrage	
2.1. Le traitement sur le résumé de données	71
2.2. Le traitement sur les données acquises	72
2.3. Le traitement du "dépassement en temps" EVB	74
3. Les instructions spécialisées MASS et HODO 2	75
3.1. L'instruction MASS	
3.2. L'instruction HODO 2	83
III. GENERATION DU LOGICIEL GESPRO	88
1. Programme	
2. Microprogramme	
3. Génération des fichiers-image	
IV. LES PERFORMANCES INTRINSEQUES DU CALCULATEUR	91
1. Le temps d'exécution des instructions spécialisées	
2. Apport des opérateurs câblés et de la microprogrammation	

V. LES RESULTATS EXPERIMENTAUX	94
1. Les temps de filtrage	
2. Exemple d'acquisition à haute intensité	95
3. Réjections des différentes coupures	97
4. Le spectre de masse	
5. Variation de la réjection en fonction de l'intensité	99
6. Variation du temps actif en fonction de l'intensité	
7. Contrôle du système d'acquisition	
7.1. Surveillance de l'acquisition par le comptage des PSI	
7.2. Tests et maintenance du système d'acquisition	102
7.3. Mesure de l'efficacité du déclenchement	
CONCLUSION	104
ANNEXES	
A. Multiplication flottante: -programmée -microprogrammée -cablée	
B. Codes: -MCU -CPE	
C. Jeu d'instructions GESPRO	
D. Organigrammes des traitements: - P1P2 - VR3R4	
REMERCIEMENTS	
REFERENCES	

Légende des figures

- Fig.1 Appareillage de l'expérience NA10
- Fig.2. Trajectoire d'une particule dans l'appareillage
- Fig.3. Bloc-diagramme du système de déclenchement câblé.
- Fig.4.1. Matrice de coïncidences.
- Fig.4.2. Matrices de décision de la logique intersextants
- Fig.5.1. Histogramme de nombre total de mots par événement à faible intensité
- Fig.5.2. Histogramme de nombre de mots hodoscopes par événement à faible intensité
- Fig.6 Tâches informatiques d'une expérience de physique des particules
- Fig.7 Bloc-diagramme du système d'acquisition classique de l'expérience NA10
- Fig.8 Bloc-diagramme de l'ensemble informatique de l'expérience NA10
- Fig.9 Tâches assurées par le système ND100/500 en acquisition
- Fig.10.1. Variation du nombre de déclenchements en fonction de l'intensité
- Fig.10.2. Histogramme du nombre total de mots par événement à intensité moyenne
- Fig.11. Histogramme des événements reconstruits en fonction du nombre de mots de l'événement
- Fig.12. Efficacité du système classique en fonction du nombre de déclenchements par cycle
- Fig.13.1. Histogramme du nombre total de mots par événement avec coupure multiplicité R
- Fig.13.2. Histogramme du nombre de mots R et V par événement avec coupure multiplicité R.
- Fig.14. Organigramme de la multiplication de 2 nombres flottants de 24 bits
- Fig.15. Bloc diagramme du système d'acquisition avec EVB
- Fig.16. Diagramme des signaux de contrôle EVB
- Fig.17. Chronogramme des signaux de contrôle EVB
- Fig.18. Structure du registre d'état
- Fig.19.1. Architecture du calculateur GESPRO
- et 19.2.
- Fig.20. Chronogramme de la resynchronisation Unité Centrale-Périphériques
- Fig.21. Codes multiplieur
- Fig.22. Format des microinstructions GESPRO
- Fig.23. Synoptique du décodage de l'instruction
- Fig.24. Synoptique du déroulement d'une instruction
- Fig.25. Organigramme du programme de filtrage NA10

- Fig.26. Table de chaînage de l'instruction MASS
- Fig.27. Organigramme de l'instruction MASS
- Fig.28. Organigramme du microprogramme MASS
- Fig.29. Tables générées par l'instruction HODO 2
- Fig.30.1. Organigramme de l'instruction HODO 2
- Fig.30.2. Organigramme du microprogramme HODO 2
- Fig.31.1. Génération du logiciel GESPRO
- Fig.31.2. Extrait du programme de filtrage GESPRO
- Fig.32.1. Histogramme du nombre d'événements biaisés en fonction de la masse du dimuon
- Fig.32.2. Résolution en masse entre la masse calculée par GESPRO et la masse calculée hors ligne
- Fig.33.1. Variation du nombre de déclenchements par cycle en fonction de l'intensité
- Fig.33.2. Variation de la réjection en fonction de l'intensité
- Fig.33.3. Variation du temps actif du système en fonction de l'intensité
- Fig.34. Histogramme du nombre d'événements signés par le déclenchement câblé en fonction du nombre de sextants touchés et temps de décision GESPRO associé.

I N T R O D U C T I O N

Cette thèse est consacrée à l'étude du système de filtrage d'évènements, utilisé par l'expérience NA10 au CERN. L'expérience a pour but l'étude des dimuons de haute masse, produits lors des collisions π^- Noyaux à diverses énergies (100-300 GeV).

Pour étudier un processus dont la section efficace est inférieure à 10^{-33} cm²/GeV et disposer d'une statistique suffisante pour obtenir des résultats significatifs, l'expérience a été conçue pour admettre des flux très élevés (supérieurs à 10^9 π^- /cycle du SPS).

Dans ces conditions, la qualité de l'expérience repose en grande partie sur l'efficacité de son système de déclenchement, une sélection non biaisée des évènements étant indispensable pour disposer d'un lot d'évènements suffisants à l'étude de la physique.

L'expérience a été réalisée dans le cadre d'une collaboration d'une vingtaine de physiciens du CERN, de l'Ecole Polytechnique, de l'ETHZ de Zürich et du CRN de Strasbourg. Approuvée en 1978, l'essentiel de la construction de l'appareillage était achevé fin 1979. Le montage et les tests ont eu lieu lors du premier semestre 1980. Une première série de prise de données a eu lieu en juin 1980 avant l'arrêt d'un an du SPS.

C'est en 1981 que nous avons complété le système de déclenchement de l'expérience par un déclenchement de second niveau.

Cette thèse décrit le matériel, le logiciel et les performances du système implanté sur NA10. Elle est divisée en deux chapitres:

- le premier décrit l'appareillage de l'expérience, les détecteurs, la logique de déclenchement, le système d'acquisition avec le filtre qui lui est associé;
- le second présente le programme de filtrage et ses instructions particulières, ainsi que les performances du système.

CHAPITRE I

I. CARACTERISTIQUES DE L'EXPERIENCE NA10

1. L'appareillage [1]

L'ensemble de l'appareillage de l'expérience NA10 forme un spectromètre à muons, conçu pour mesurer avec une grande statistique et une bonne résolution en masse, la production de dimuons à partir d'un faisceau de pions à haute intensité (jusqu'à $6 \times 10^9 \pi$ par cycle faisceau)

La figure 1 représente l'appareillage du point de vue mécanique. Il se compose essentiellement des éléments suivants :

1.1. La cible

La cible est constituée d'un barreau de tungstène de 18 mm de diamètre et 58 mm de longueur. Elle est montée sur un barillet qui peut être équipé de 4 cibles différentes et que l'on peut commander à distance.

1.2. L'absorbeur

L'absorbeur composé de tungstène, graphite et fer a pour rôle d'absorber la partie du faisceau qui n'a pas interagi avec la cible et les particules qui ne sont pas des muons.

1.3. Les quatre hodoscopes hexagonaux R_1 à R_4

Leur géométrie est adaptée à la géométrie particulière de l'aimant du spectromètre.

Chaque hodoscope est formé de six sextants identiques, orthogonaux à l'axe du faisceau. Chaque sextant est constitué de lattes de scintillateurs parallèles entre elles, reliées à des photomultiplicateurs par l'intermédiaire de guides en plexiglas.

32 compteurs par sextant pour les hodoscopes R_1, R_2 et R_4 , 23 compteurs par sextant pour l'hodoscope R_3 , donnant un total de 714 voies.

Les deux hodoscopes R_1 et R_2 sont situés devant l'aimant. L'hodoscope R_1 est l'homothétique de l'hodoscope R_2 par rapport au centre de la cible. Dans chaque sextant, la largeur des compteurs est en progression géométrique. Cette géométrie particulière permet, en associant les compteurs correspondants de R_1 et R_2 de ne sélectionner que les traces qui pointent sur la cible et de mesurer l'angle qu'elles forment avec l'axe du faisceau.

Les deux hodoscopes R_3 et R_4 sont situés derrière l'aimant. Leurs sextants sont composés de lattes de scintillateurs jointives de largeurs identiques. Ils ont pour rôle la mesure de la déviation des traces définies par $R_1 R_2$ après passage de la particule dans l'aimant.

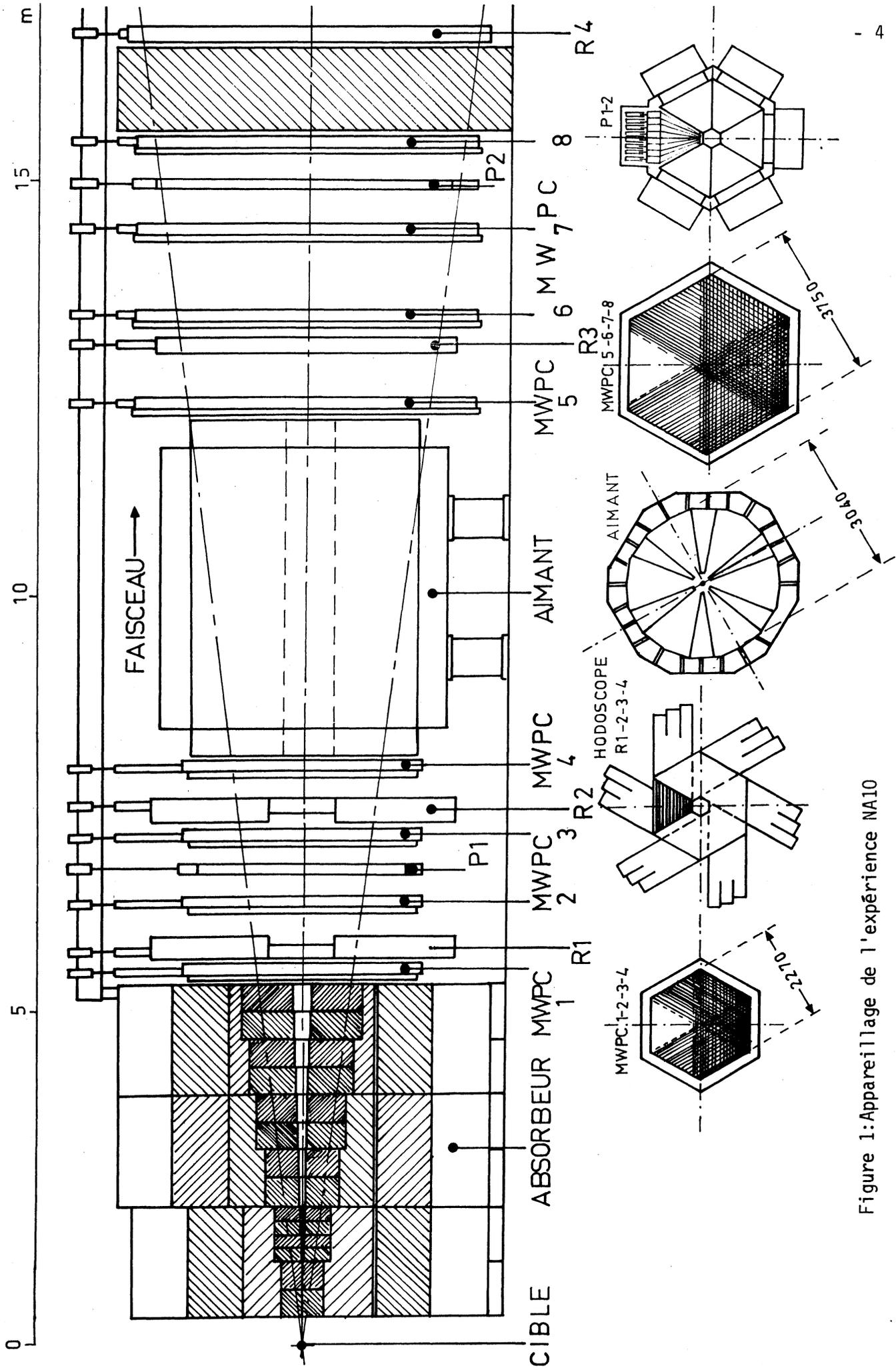


Figure 1:Appareillage de l'expérience NA10

1.4. L'aimant

L'aimant du spectromètre est un aimant toroïdal formé de six sextants identiques, symétriques par rapport à l'axe du faisceau. Il comprend six bobines et six secteurs de fer, et produit un champ radial de l'ordre de 3 Tesla-mètres. Il fonctionne en mode pulsé, en synchronisme avec le faisceau. L'intensité du courant et sa polarité sont programmables. L'intensité peut monter jusqu'à 10 000 ampères pendant la phase active du faisceau.

Sous l'effet du champ radial créé par l'aimant, les particules qui le traversent, sont défléchies dans un plan φ contenant l'axe du faisceau (cf fig.2). Comme le champ magnétique varie en $1/R$ (R est la distance entre l'impact de la particule et l'axe de l'aimant), la déflexion subie par la particule est proportionnelle à son impulsion transverse. Cette propriété est utilisée par le système de déclenchement pour sélectionner les événements à partir de la détermination des impulsions transverses de deux muons.

1.5. Les deux hodoscopes P1 et P2

Les hodoscopes P1 et P2 sont situés de part et d'autre de l'aimant. Ils sont formés de six sextants, contenant chacun huit scintillateurs reliés à des photomultiplicateurs, et servent à affiner la sélection des bons événements qui traversent l'appareillage.

1.6. Les huit chambres proportionnelles multifils PC1 à PC8

Ces huit chambres sont réparties pour moitié en amont et en aval de l'aimant. Chaque chambre est composée de trois plans de fils perpendiculaires à l'axe du faisceau, chaque plan étant parallèle à une direction formant un angle de 60° par rapport aux deux autres.

2. Le système de déclenchement [2]

2.1. Principe

La figure 3 donne le bloc-diagramme du système de déclenchement câblé. L'expérience ne s'intéresse qu'aux dimuons de haute masse créés dans la cible. L'absorbeur placé derrière la cible effectue un premier filtrage des muons, mais il est transparent:

- aux muons produits dans l'absorbeur,
- au halo de muons qui accompagne le faisceau de pions,
- aux désintégrations des pions entre la cible et l'absorbeur,
- aux gerbes électromagnétiques et hadroniques qui se développent dans l'absorbeur.

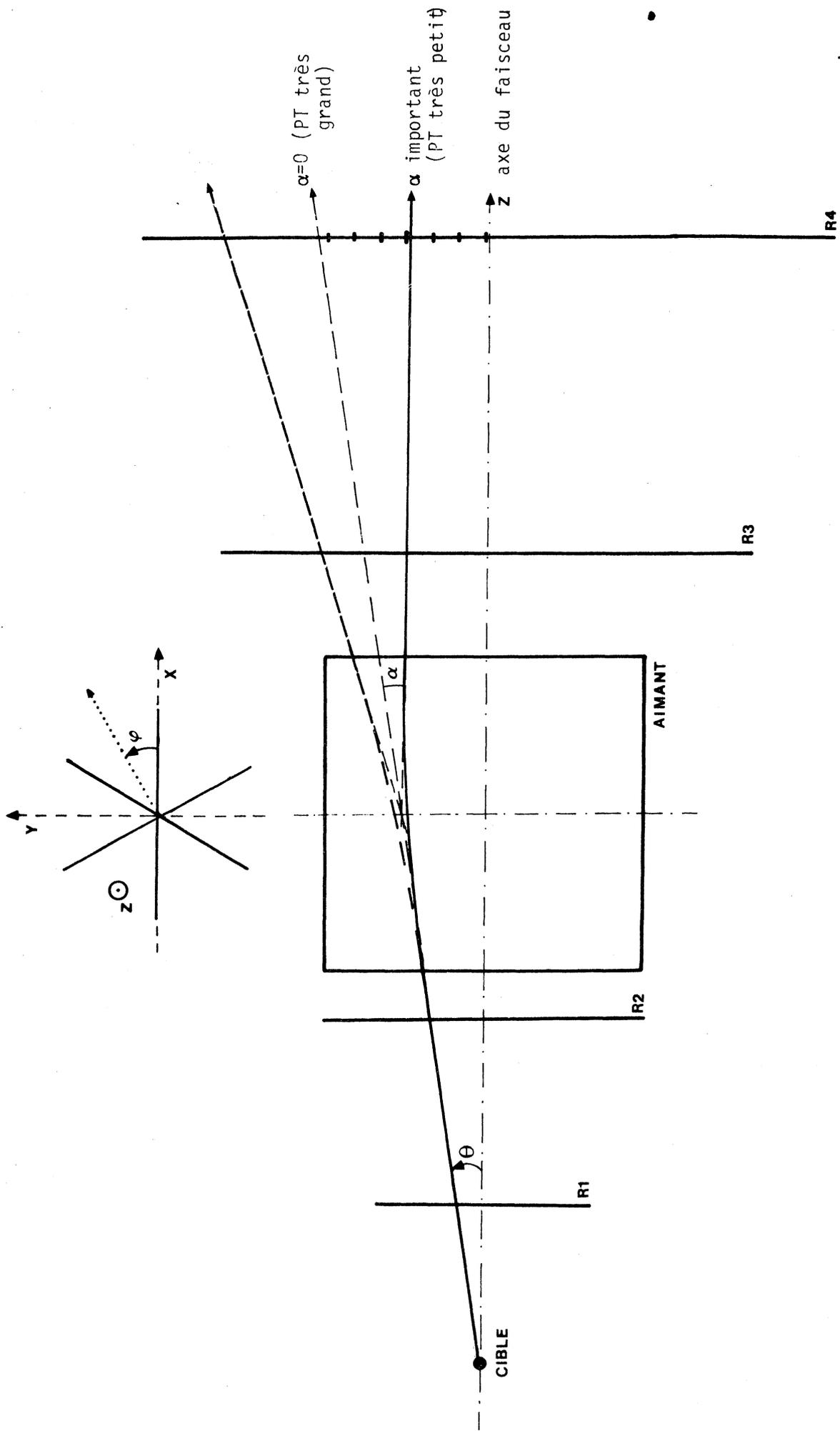


Figure 2 : Trajectoire d'une particule dans l'appareillage

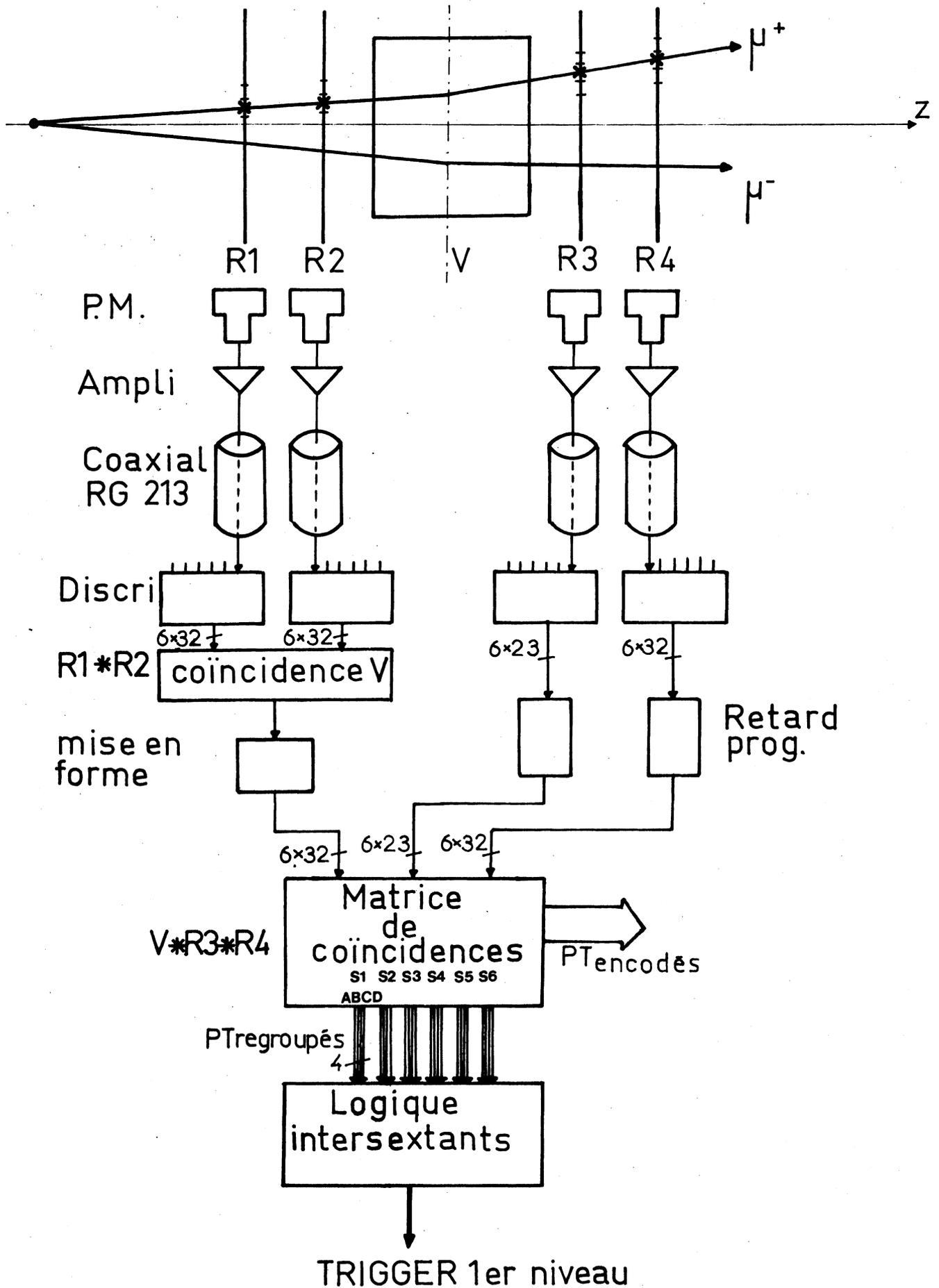


Figure 3: Bloc-diagramme du déclenchement câblé

La mise en coïncidence V des signaux issus des hodoscopes R1 et R2, permet une bonne réjection de ces événements parasites en ne sélectionnant que les traces qui pointent sur la cible.

La déflexion de la trace fournie par les compteurs de l'hodoscope R4, associée à l'angle de production fourni par la coïncidence V, permet de mesurer l'impulsion transverse du muon, qui est inversement proportionnelle à sa déflexion.

La masse du dimuon est approximée par la somme des moments transverses des deux muons:

$$M = P_{T1} + P_{T2}$$

Si la masse estimée est supérieure à la coupure programmée, le système de déclenchement émet un signal d'acquisition.

En principe la coïncidence V*R4 est produite par un muon de moment transverse P_T , issu de la cible, qui traverse l'appareillage. Mais une coïncidence fortuite entre par exemple, un muon issu de la cible validant R1*R2, et un halo de muons frappant simultanément R4, peut simuler un bon événement. Pour réduire ce taux de fortuites, la coïncidence V*R4 est validée par les compteurs de l'hodoscope R3. La trace d'un muon traversant l'appareillage doit alors satisfaire à la condition d'alignement V*R3*R4 pour être reconnue par le système de déclenchement.

2.2. La logique associée

2.1.1. La coïncidence V

La coïncidence V est réalisée par une unité de logique rapide. L'hodoscope R1 étant l'homotétique de l'hodoscope R2 par rapport au centre de la cible, la coïncidence ne s'établit que si une particule traversant le ième scintillateur d'un sextant de R1, traverse le ième scintillateur du même sextant de R2.

2.1.2. La matrice de coïncidences

La matrice de coïncidences est formée de 192 modules, à raison d'un V par module, de 6 sextants et de 32 V par sextant. A chaque valeur V, correspond une matrice (R3xR4) des coïncidences V*R3*R4 autorisées. Adressée par un couple (V, R3, R4), elle ne délivre la valeur de l'impulsion transverse associée, que si la coïncidence est autorisée (fig. 4-1).

Pour chaque sextant, la matrice peut émettre 4096 valeurs d'impulsion, délivrées sous deux formes:

- 32 signaux de P_T encodés (P_{TE}):

Pour chaque compteur V, les compteurs R4 sont renumérotés de 0 à 31, dans l'ordre décroissant des valeurs d'impulsion transverse associée. En cas de conflit entre plusieurs coïncidences V*R4 pour un même V, c'est le numéro de compteur R4 donnant lieu au plus grand moment transverse qui est conservé et délivré par la matrice. Nous verrons dans un chapitre suivant que cette information est nécessaire au calcul de masse effectué par le calculateur GESPRO.

- 4 signaux de P_T regroupés (P_{TR}):

Les 32 valeurs possibles de P_T sont regroupées en 4 intervalles, tous V confondus, notés A, B, C et D, correspondant à 4 zones d'impulsion transverse:

A	$P_T > 4 \text{ GeV/c}$
B	$2 \leq P_T < 4 \text{ GeV/c}$
C	$1.4 \leq P_T < 2 \text{ GeV/c}$
D	$0.6 \leq P_T < 1.4 \text{ GeV/c}$

Ces valeurs approchées permettent à la logique intersextants, d'approximer la masse du dimuon sur un nombre réduit de combinaisons. Les intervalles sont choisis de façon à avoir une bonne précision aux alentours de la coupure en masse ($\approx 4 \text{ GeV}$), prévue dans la logique intersextants. En cas de conflit entre plusieurs valeurs de P_T dans un même sextant, un encodeur de priorité favorise la plus grande valeur pour éviter toute perte d'évènements de haute masse.

2.2.3. La logique intersextants

La logique intersextants, grâce aux informations fournies par la matrice de coïncidences, reconnaît la présence de plusieurs muons, définit la région de masse à laquelle appartient le ou les dimuons, et émet un signal d'acquisition si la masse est supérieure à la coupure de masse programmée.

Aux 6 sextants correspondent 15 combinaisons intersextants. A chaque combinaison intersextants (S_i, S_j) correspond une matrice de décision ($P_{TRi} \times P_{TRj}$) (cf fig. 4). La masse du dimuon étant définie par la somme des moments transverses de 2 muons, à chaque couple (P_{TRi}, P_{TRj}) correspond un intervalle de masse.

La matrice associée à chacun de ces couples, une décision:

- lancement de l'acquisition pour les dimuons de hautes masses (H) et de masses intermédiaires (I),
- rejet pour les faibles masses (L).

Les 15 combinaisons sont réparties en trois groupes :

- sextants adjacents
- sextants alternés
- sextants opposés

auxquels correspondent 3 types de matrice de décision, permettant de ne pas appliquer obligatoirement les mêmes critères de déclenchement aux dimuons mesurés dans des sextants adjacents, opposés ou alternés (cf. figure 4-2).

La logique intersextants est réalisée en logique ECL.

Le temps nécessaire entre le passage de la particule et la décision est d'environ 400 ns.

3. Le système de mémorisation^[3]

L'information des détecteurs représente plus de 21000 voies à mémoriser. Cette information est stockée dans une mémoire rapide associée aux détecteurs, le RMH (Receiver Modules Hybride), constituée de 660 modules de 32 mots de 16 bits, et organisée en 9 branches : une branche par chambre et une branche contenant les données hodoscopes et les informations fournies par la matrice de coïncidence et la logique intersextants.

Chaque début de branche est désigné par un mot-clé programmable écrit dans la mémoire par des modules spéciaux, les simulateurs d'encodeurs de châssis (CES).

Le RMH compacte l'information. Pour chaque événement, il ne transmet que l'adresse des compteurs et des fils touchés, évitant un transfert et une capacité mémoire prohibitifs en aval du RMH. La longueur du bloc de données est donc variable et dépend de la nature de l'événement sélectionné.

Le temps de transfert typique de la mémoire RMH vers un système de lecture approprié est de 150 ns par mot de 16 bits.

4. Structure de l'information à acquérir

Les histogrammes 1 et 2 figure 5, montrent, qu'à faible intensité la taille moyenne d'un événement est d'environ 190 mots dont 40 mots hodoscopes. Cette information est structurée comme suit :

- 36 mots d'en-tête caractérisant l'événement
- 1 marqueur de début de branche hodoscope
- 1 code de la combinaison de sextants touchés par le dimuon qui a donné lieu au déclenchement
- 1 marqueur indiquant si l'événement est non biaisé

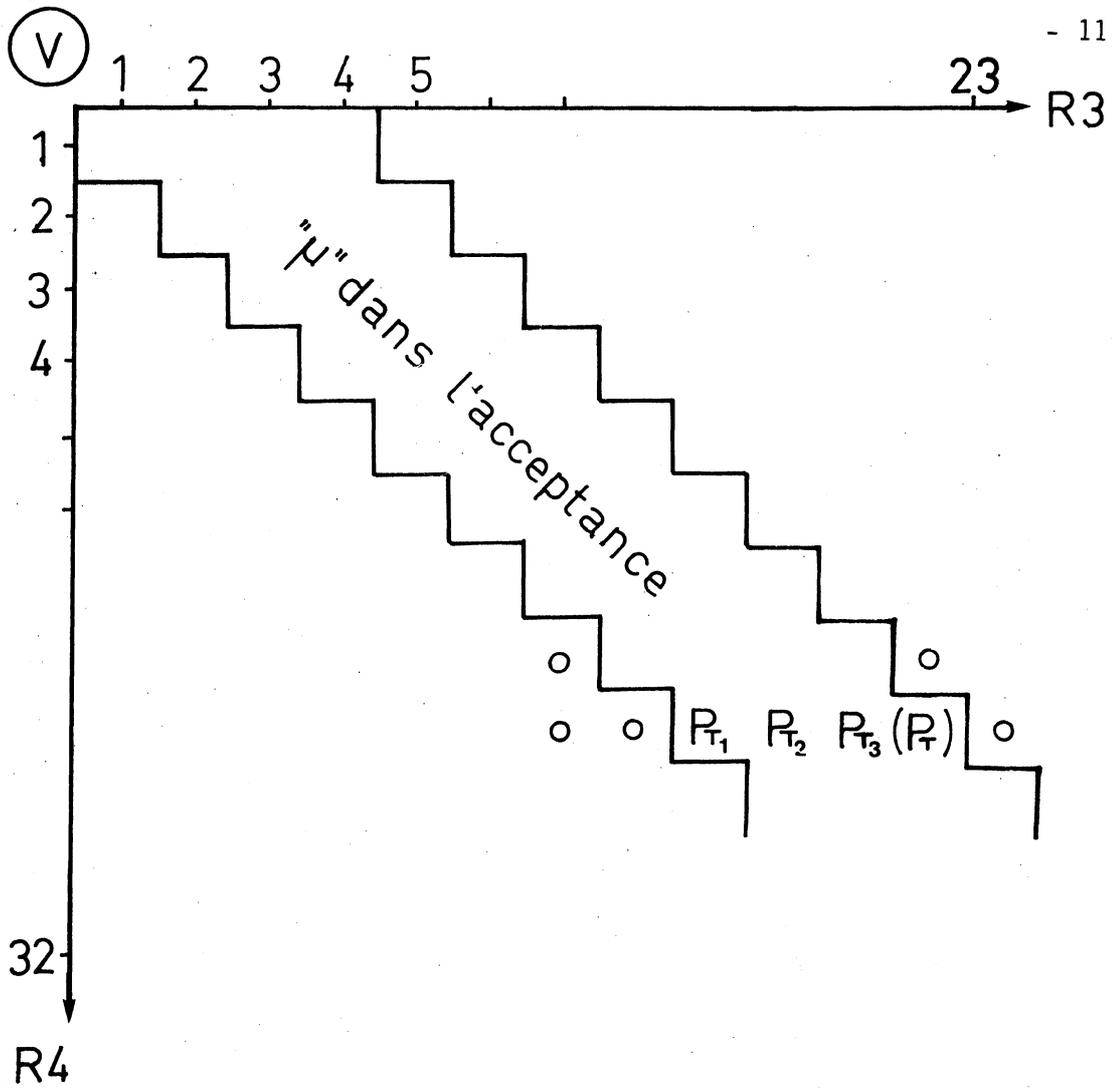


Figure 4-1: Matrice de coïncidence V*R3*R4

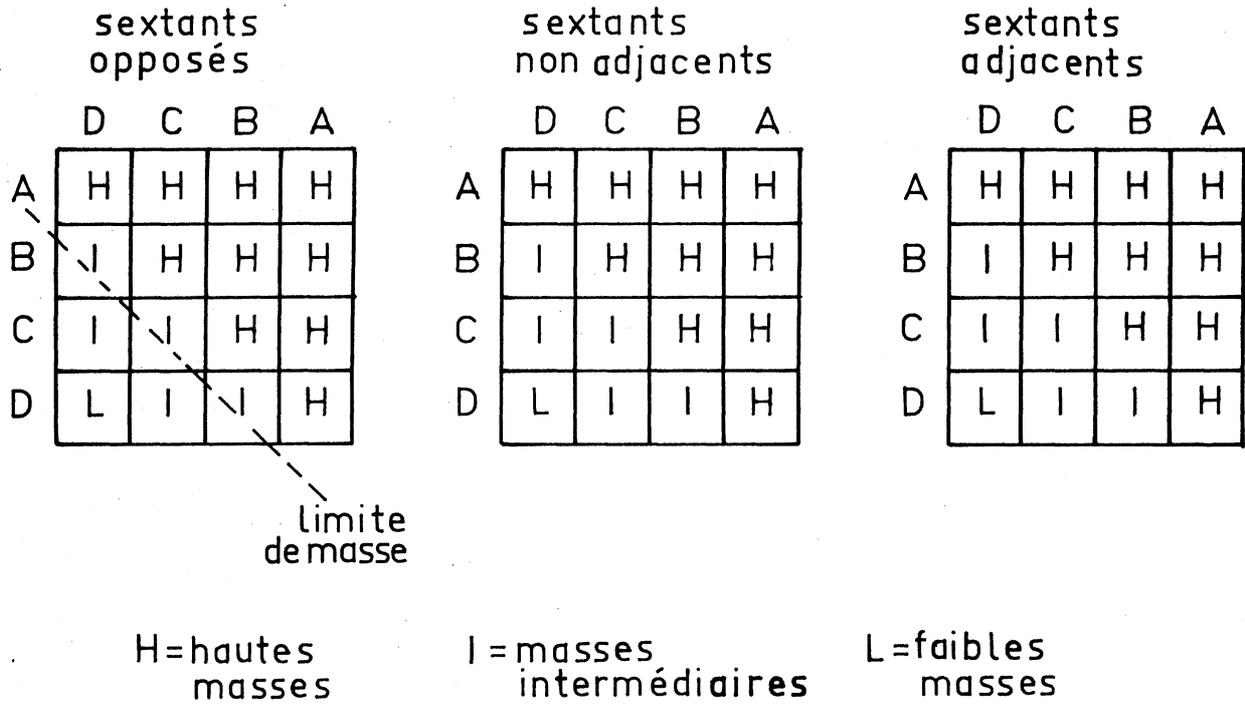


Figure 4-2: Matrices de décision de la logique intersextants

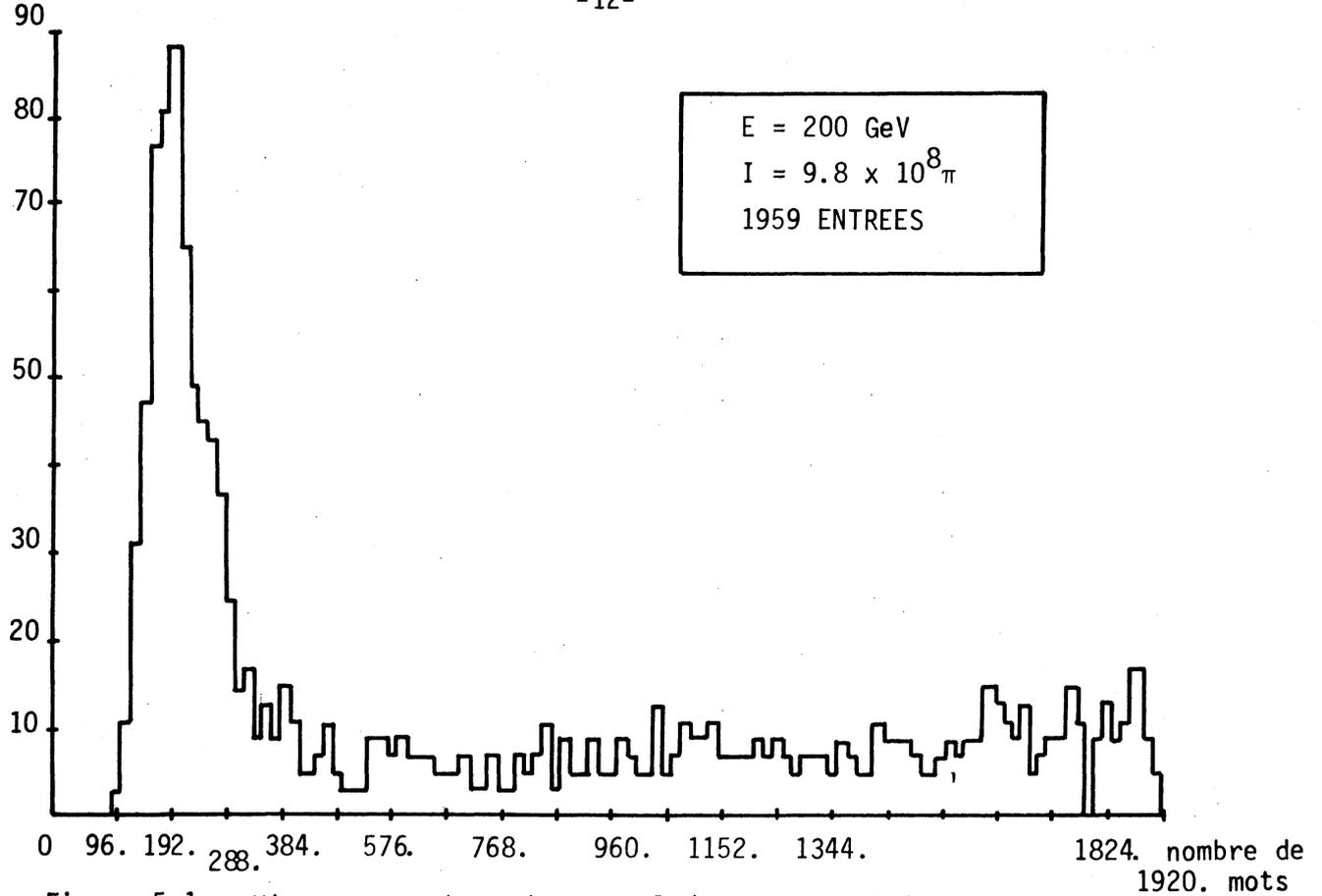


Figure 5.1. : Histogramme du nombre total de mots par événement à faible intensité (623 dépassements)

nombre d'evts

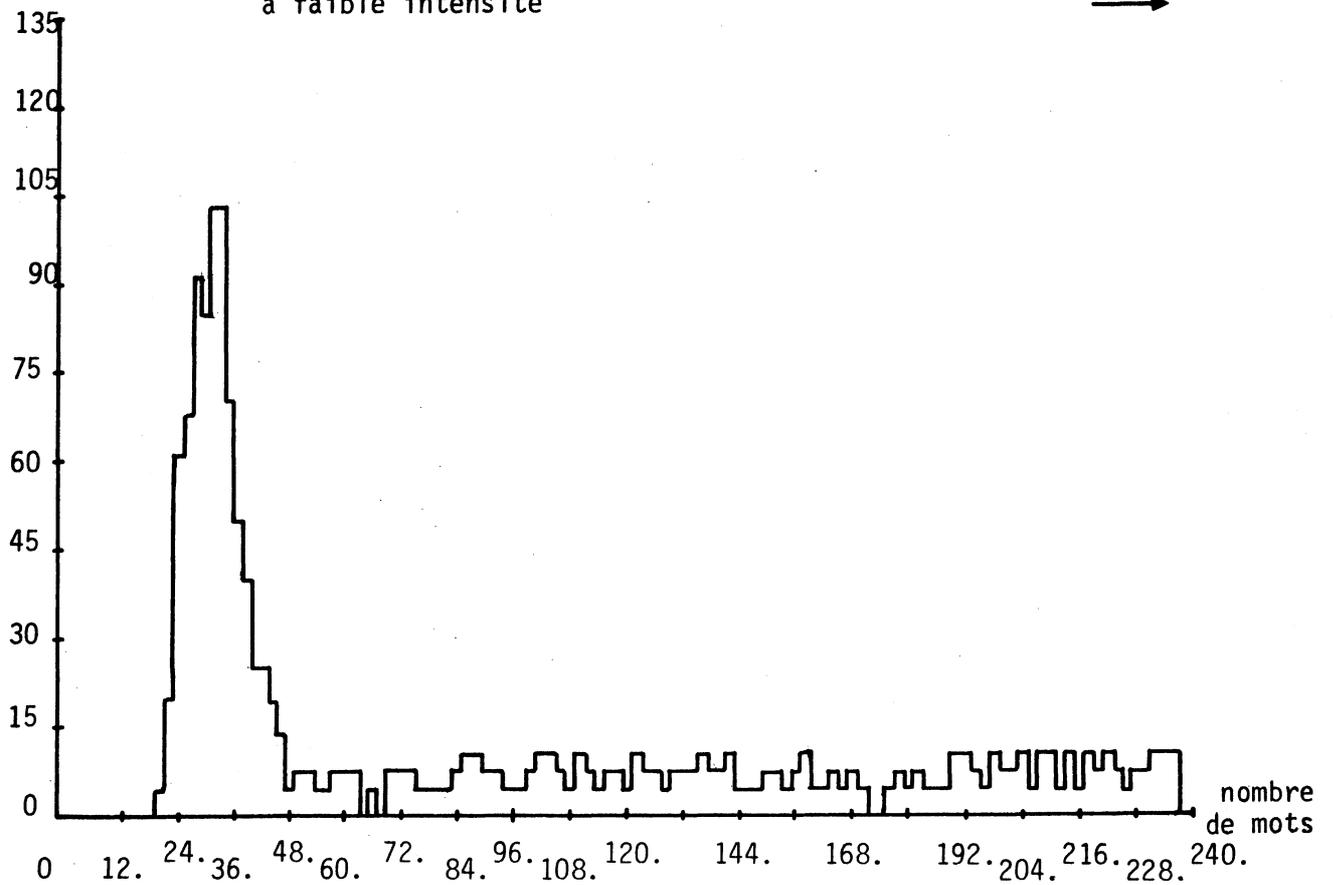


Figure 5.2. : Histogramme du nombre de mots hodoscope par événement à faible intensité (691 dépassements)

- 1 marqueur indiquant si la masse du dimuon approximée par la matrice se trouve dans la zone de masse intermédiaire PSI
- 2 mots P_T regroupés
- 8 adresses compteurs R ($2R_1, 2R_2, 2R_3, 2R_4$) et 2 adresses compteurs V minimum
- 4 adresses compteurs P ($2P_1, 2P_2$)
- 2 coïncidences P_1P_2
- 96 mots chambres minimum (1 dimuon activant 2 fils par plan, à raison de 3 plans par chambre et de 8 chambres), en 8 groupes de données séparés par 8 marqueurs de début de chambre.

II. LE SYSTEME D'ACQUISITION [4][6]

1. Les fonctions du système d'acquisition

La figure 6 schématise les fonctions assurées par l'acquisition dans une expérience de physique des particules utilisant des compteurs. Les principales fonctions sont :

- déclenchement
- acquisition des données et mémorisation sur bande magnétique
- contrôles et tests automatiques des détecteurs et de l'électronique de l'expérience
- prétraitement et filtrage des données permettant une reconstruction partielle de l'événement, afin de détecter d'éventuelles erreurs de fonctionnement ou d'alignement des capteurs.

Les tâches de contrôles et d'acquisition sont généralement confiées à des ordinateurs banalisés. On dispose ainsi :

- du système fourni par le constructeur
- de la maintenance constructeur
- de la facilité d'utilisation qu'apporte l'emploi de langages de programmation évolués.

Mais la nécessaire standardisation des différents composants matériels du système d'acquisition, ne permet de connecter le miniordinateur à l'expérience qu'à travers un système d'interconnexion standard (CAMAC) qui rend les opérations d'entrées-sorties lentes vis-à-vis du temps de cycle des compteurs (en technologie ECL), actuellement mis en oeuvre sur les expériences de physique des particules. (cf. Le système d'acquisition classique NA10 - paragraphe 2).

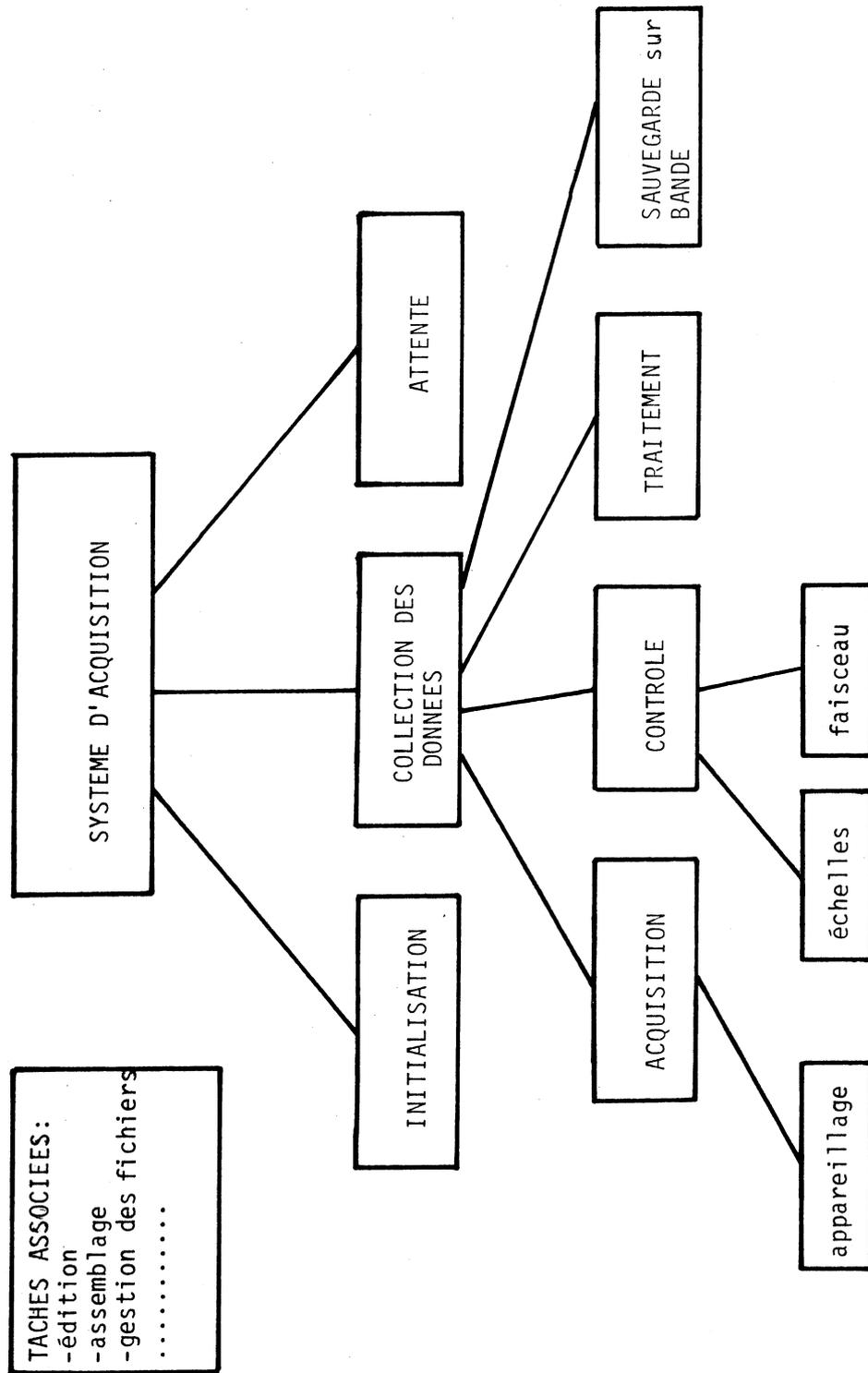


Figure 6 : Tâches informatiques d'une expérience de physique des particules

[Un nouveau standard d'interconnexion FASTBUS améliorant les performances du transfert a été élaboré, mais les premières réalisations n'apparaissent qu'actuellement.]

Une adaptation des flux entre compacteurs et mémoire d'accumulation est donc nécessaire. Pour l'obtenir, on peut augmenter la puissance d'acquisition du système, en distribuant cette acquisition sur plusieurs miniordinateurs, mais le nombre de machines à mettre en oeuvre pour obtenir un masquage significatif, rend cette méthode extrêmement onéreuse.

On peut également modifier la structure d'un miniordinateur courant, mais cette modification est limitée en efficacité et pose des problèmes de maintenance.

L'adaptation du flux s'obtient également en augmentant la puissance de réjection du système de sélection, de façon à réduire le volume de l'information à mémoriser. Cette puissance s'obtient en introduisant dans la chaîne d'acquisition, un système informatique au niveau de la sélection. Contrairement au contrôle et à l'acquisition, les fonctions demandées par la sélection des événements sont spécifiques à l'expérience et les temps d'exécution doivent être les plus courts possibles, ce qui implique la mise en oeuvre de processeurs spécialisés (cf. le système d'acquisition NA10 avec EVB:paragraphe III).

2. Le système d'acquisition "classique" de l'expérience NA10

2.1. Architecture

La figure 7 donne le bloc diagramme du système d'acquisition "classique". Les données des détecteurs et du système de déclenchement sont mémorisées dans la mémoire RMH par la signature d'un événement. A travers un interface accessible par le système d'interconnexion CAMAC ^[5], le contenu de la mémoire RMH est transmis en mode Accès Direct Mémoire (DMA) à la mémoire d'un couple de miniordinateurs ND 100/500, qui traite ces données et les sauvegarde sur bande magnétique. L'analyse est effectuée ultérieurement sur un gros ordinateur.

(CERN ou laboratoires associés à l'expérience).

2.2. Caractéristiques du transfert CAMAC

L'interface RMH-CAMAC est placée dans la Branche 0 du système CAMAC et fonctionne en CAMAC-Mode ^[5], permettant une vitesse de transfert DMA de 1.5 μ s par mot de 16 bits.

2.3. L'ensemble informatique attaché à l'expérience :

2.3.1. Le matériel

La figure 8 donne le bloc diagramme de l'ensemble informatique. L'acquisition et le traitement des données en ligne, ainsi que le contrôle des détecteurs sont assurés par un système NORSK DATA ND 100/500. Le ND500 est un processeur arithmétique rapide de 32 bits. Il nécessite un ordinateur associé pour toutes les opérations d'entrées-sorties. Sa mémoire de contrôle est volatile, et doit être chargée à partir d'un fichier de microcodes avant toute activation. L'ensemble des opérations entrées-sorties, de gestion de fichiers, d'édition, etc... est assuré par un miniordinateur ND100 de 16 bits géré par un système d'exploitation SINTRAN et capable d'exécuter des programmes en temps réel.

Les deux processeurs se partagent une mémoire multi-accès de 1 Moctets. Deux unités de bandes magnétiques 1600 bpi et un disque de 75 Moctets complètent l'ensemble matériel. L'ordinateur ND100 est relié au réseau CERNET (interconnexion des ordinateurs du CERN) et aux ordinateurs de contrôle de faisceau.

2.3.2. Le logiciel

La figure 9 schématise l'ensemble des tâches assurées par le système.

a) Le programme d'acquisition :

Le programme d'acquisition est activé par une interruption externe de début de cycle-faisceau. Le canal DMA est initialisé et autorise le transfert des données de la mémoire RMH vers un des blocs de la mémoire NORD. Lorsqu'un bloc est plein, un nouveau lancement de transfert DMA est nécessaire pour adresser un autre bloc. Huit transferts DMA peuvent être lancés par cycle, pour une acquisition maximale de 182 K mots. Cette acquisition est suspendue, soit lorsque la mémoire-tampon est pleine, soit par une deuxième interruption externe de fin de cycle (EOB).

b) Le programme de traitement :

Entre deux cycles-faisceau le processeur ND500 est activé et exécute les tâches suivantes :

- décodage de l'information contenue dans la mémoire tampon pour trouver le début des événements marqué par un drapeau, car le RMH compacte l'information sans aucun traitement et la transmet sans discontinuité. Le programme découpe cette information, événement par événement et ajoute avant chaque événement un en-tête caractéristique de l'événement (position dans le cycle, type de l'événement,...)

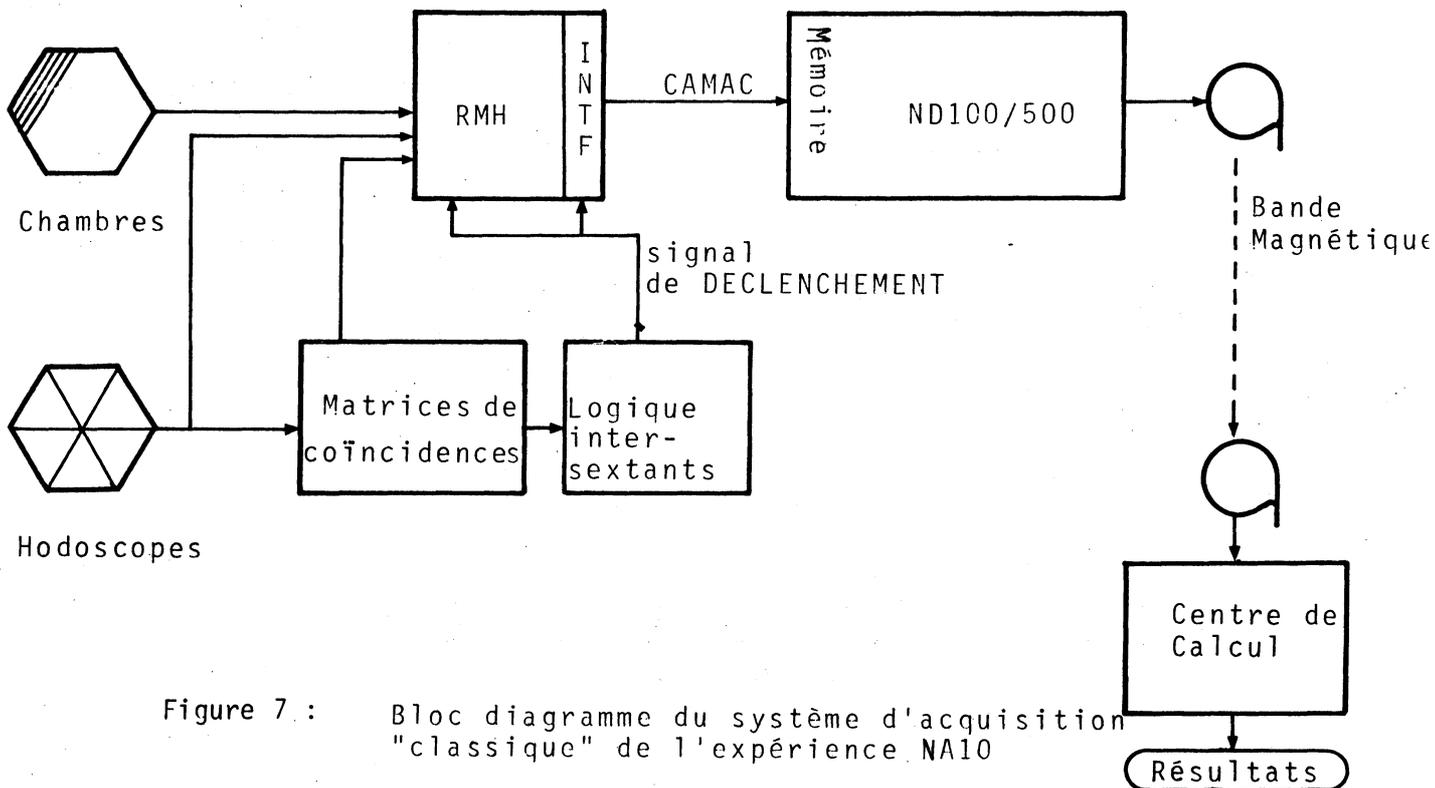


Figure 7 : Bloc diagramme du système d'acquisition "classique" de l'expérience NA10

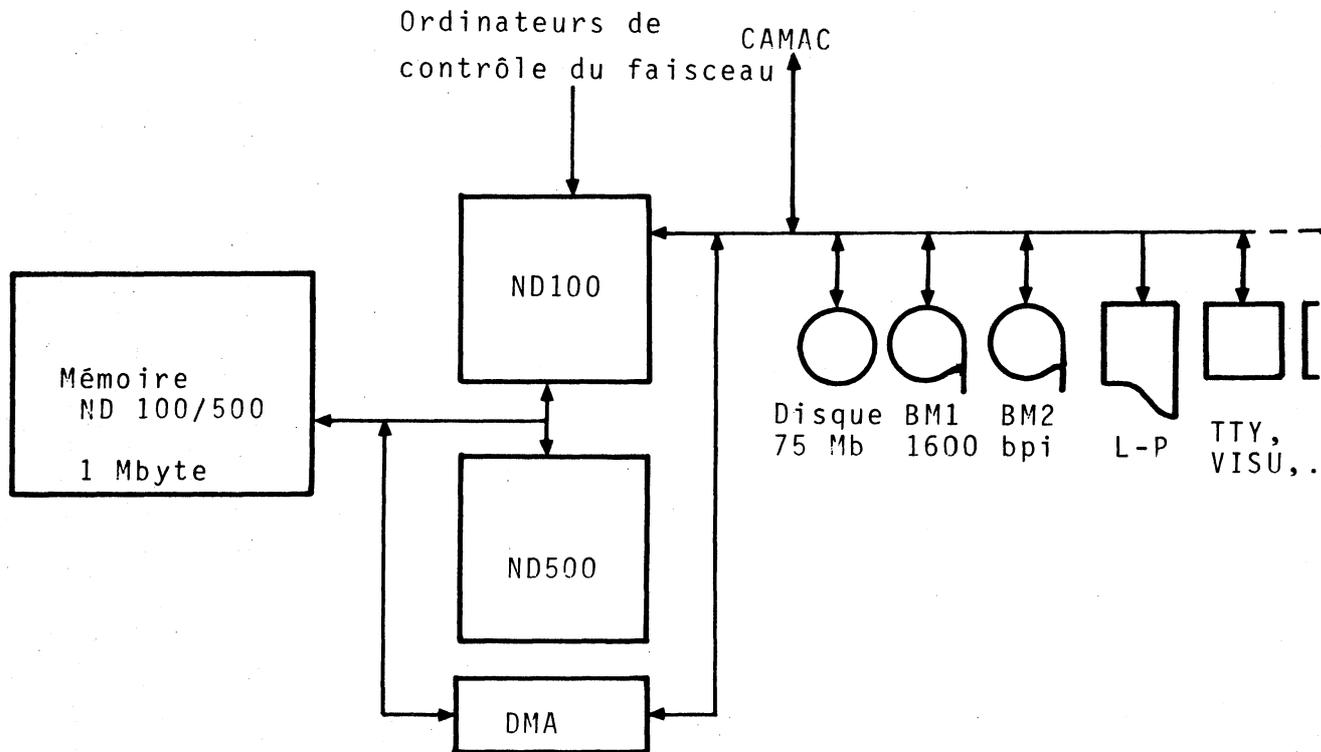
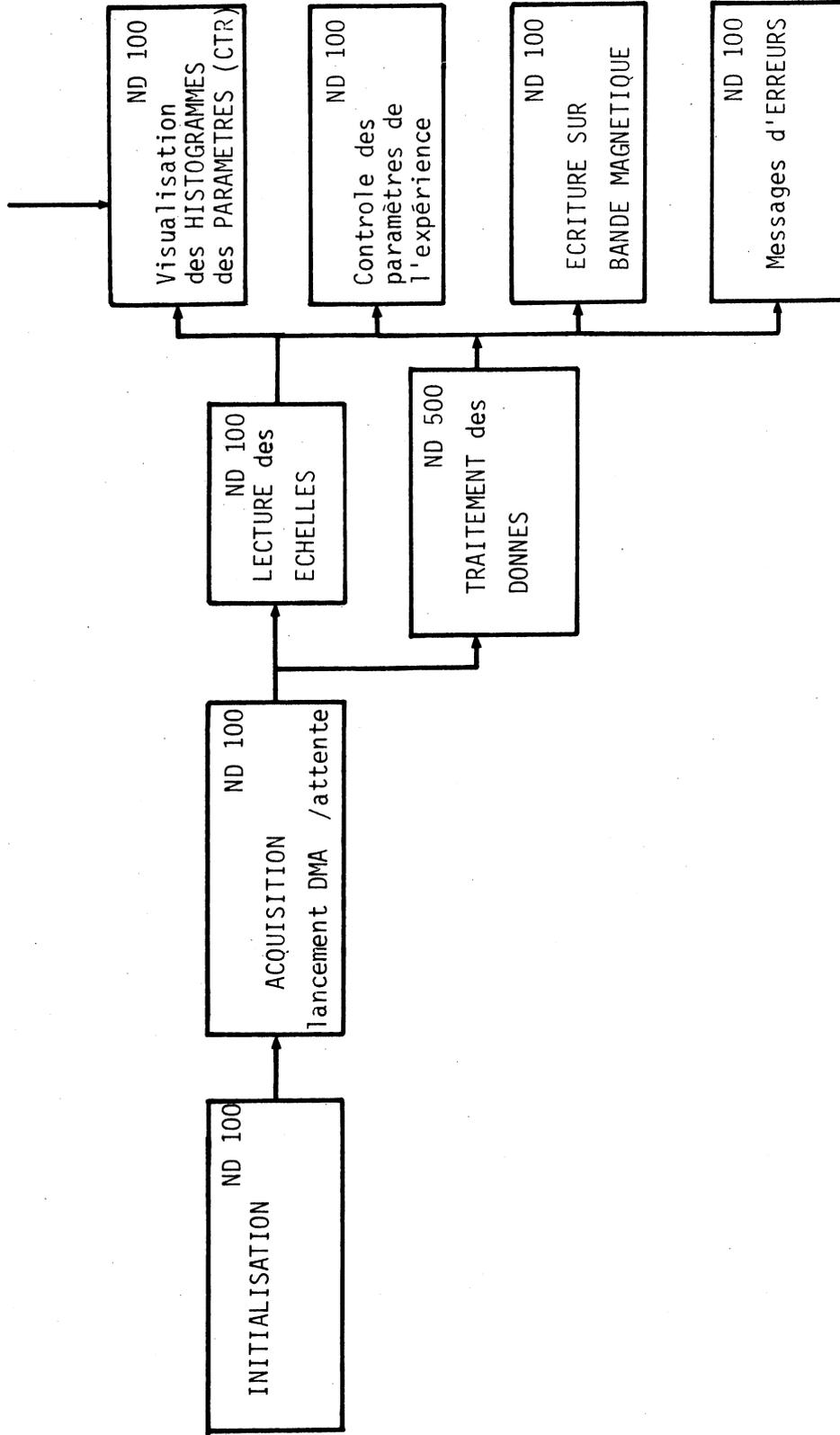


Figure 8 : Bloc diagramme de l'ensemble informatique de l'expérience NA10

COMMANDES OPERATEUR



FAISCEAU

Figure 9: Tâches assurées par le système ND100/500 en acquisition

- transfert de cette information sur bande magnétique (temps de transfert moyen : 9 μ s/octet)
- histogrammation des paramètres physiques concernant les chambres, les hodoscopes, la taille des événements,....

Simultanément, et sans ralentissement du traitement ND500, le ND100 contrôle les paramètres du faisceau et visualise les paramètres et les histogrammes demandés par l'utilisateur, ainsi que les éventuels messages d'erreurs transmis par le traitement ND500.

3. Limites du système d'acquisition classique NA10

Les limites du système d'acquisition classique NA10 sont liées :

- aux caractéristiques matérielles de la chaîne d'acquisition
- à la structure informatique de l'expérience

3.1. Limites liées aux caractéristiques matérielles

3.1.1. Manque de sélectivité du déclenchement :

Les corrélations géométriques et temporelles simples du déclenchement décrites au paragraphe 2 ne sont efficaces que pour les événements de faible multiplicité.

Lorsque l'intensité du faisceau incident augmente, la multiplicité des événements augmente et les déclenchements dus à des coïncidences fortuites se multiplient (fig.10-1).

De même, lorsque des gerbes hadroniques issues de l'absorbeur allument de nombreux compteurs, le déclenchement perd toute sélectivité.

La courbe, figure 10.2 montre, qu'à haute intensité, le pic des bons événements est noyé dans le bruit de fond.

Ce manque de sélectivité entraîne :

- une rapide saturation de la mémoire tampon du miniordinateur par des événements de grande taille (supérieurs à 1 k mots) et qui ne contiennent qu'une faible proportion d'événements restructurables (fig.11).
- plusieurs centaines d'heures de calcul hors ligne pour sortir les quelques pourcents de bons événements de la masse des événements acquis.

Nombre de déclenchements /
cycle faisceau

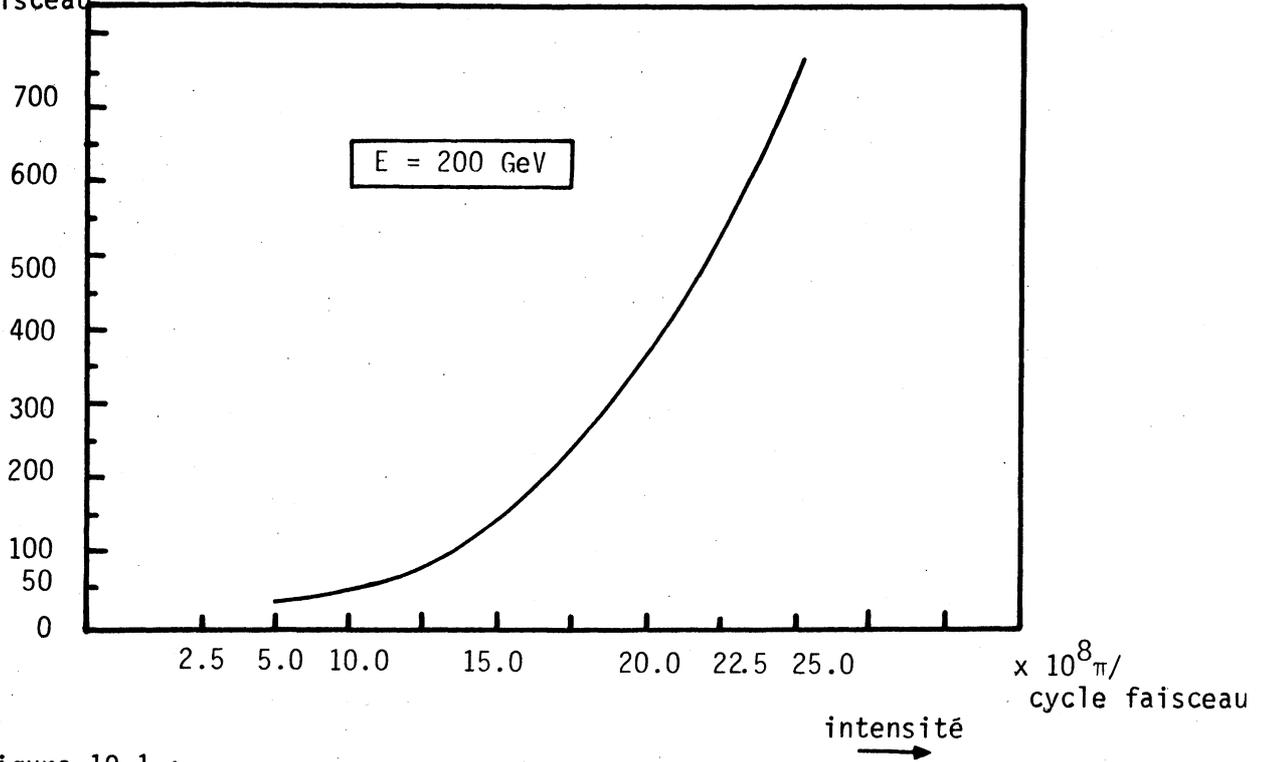


Figure 10-1 :
Variation du nombre de déclenchements en fonction de l'intensité

Nombre de déclenchements/
cycle faisceau

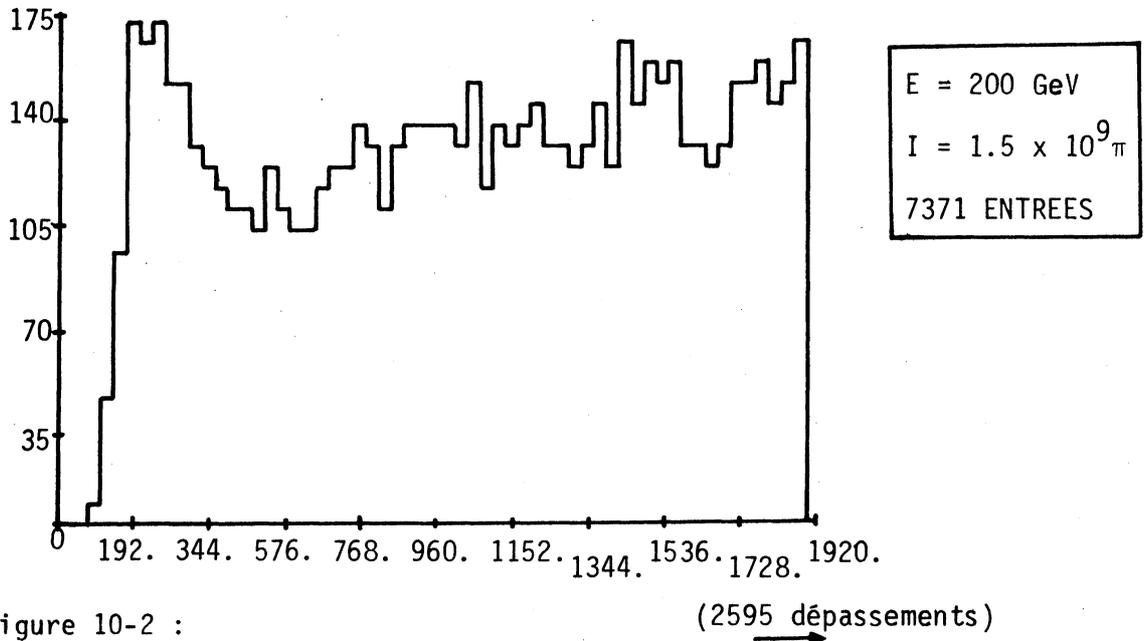


Figure 10-2 :
Histogramme du nombre total de mots par événement
à intensité moyenne.

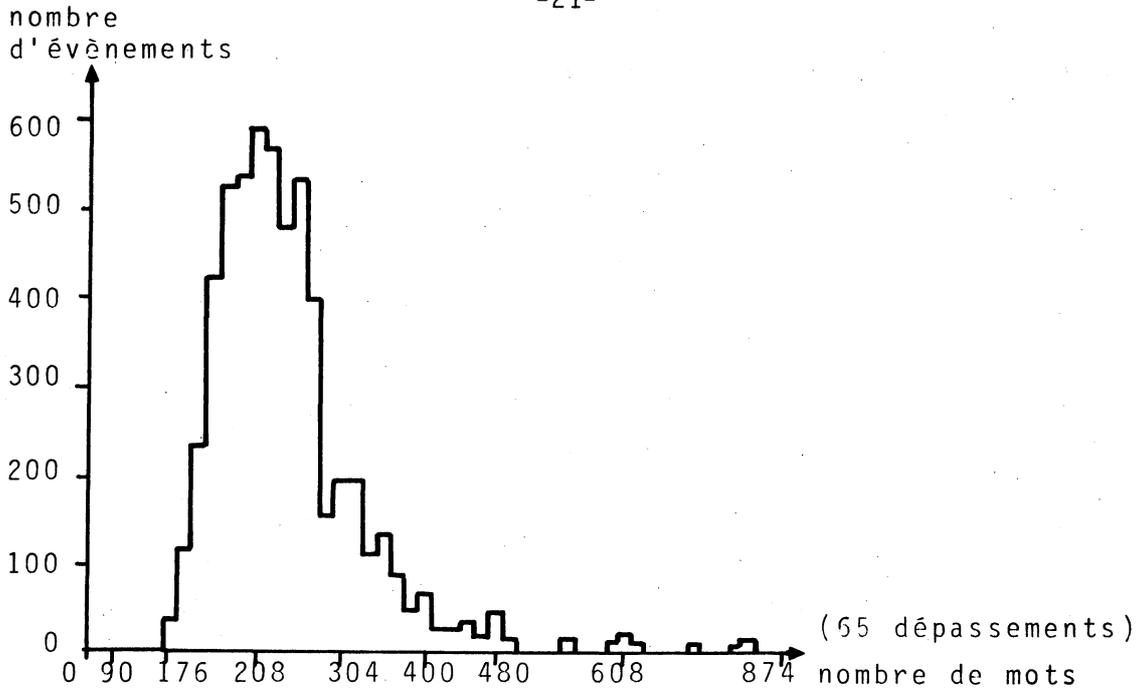


Figure 11 : Histogramme des évènements reconstruits en fonction du nombre de mots de l'évènement

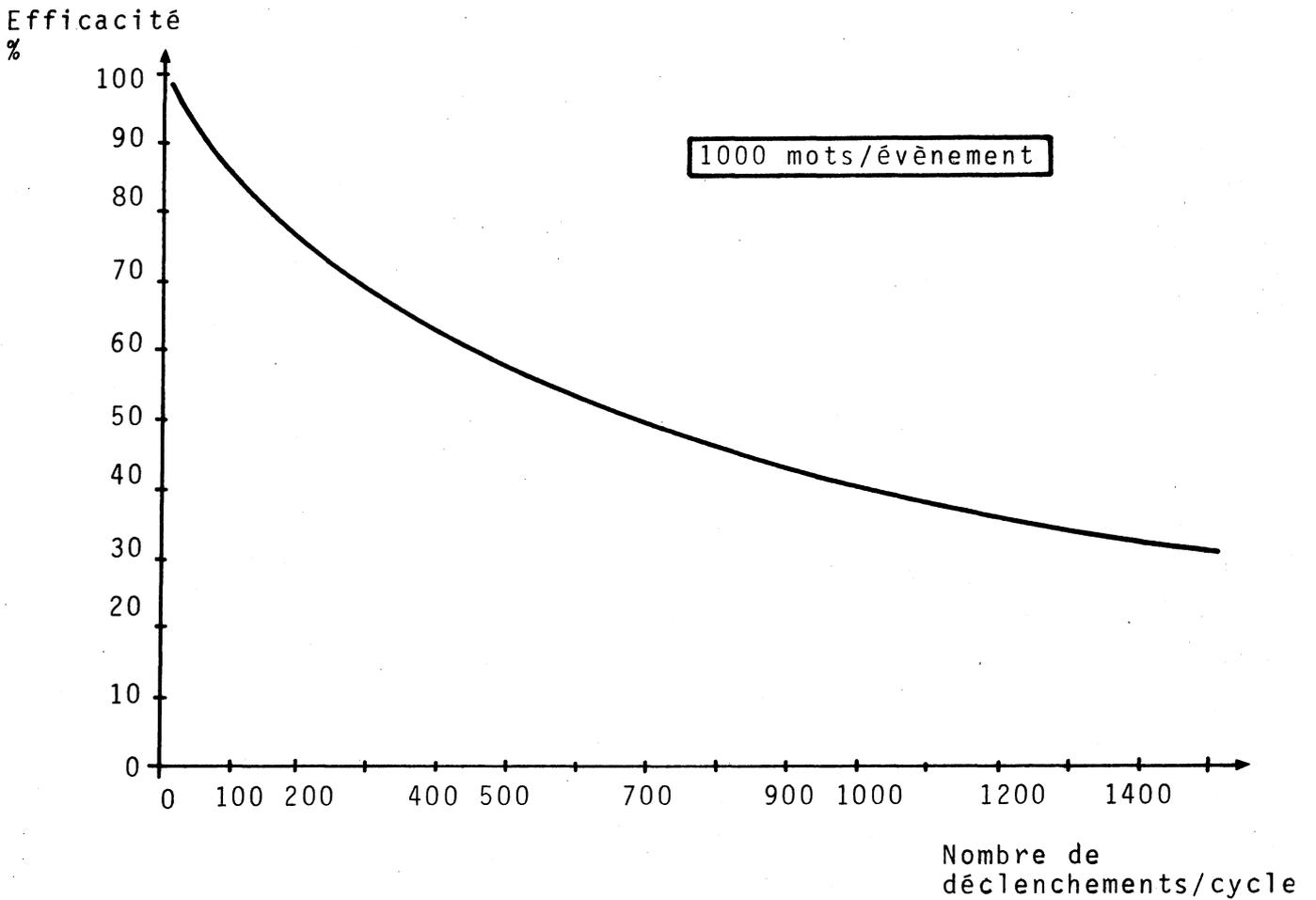


Figure 12 : Efficacité du système "classique" en fonction du nombre de déclenchements par cycle

La sélectivité peut être améliorée grâce à l'utilisation d'un veto multiplicité appliqué au système de déclenchement.

Ce veto a deux inconvénients :

- La non linéarité de l'électronique ne permet pas de faire une coupure précise sur la multiplicité (fig.13.1 et 13.2.).
- Ce veto étant appliqué au niveau de déclenchement, son inefficacité est difficilement contrôlable en permanence.

3.1.2. Le temps de réponse RMH-NORD

La plus importante limite du système classique provient des temps de réponse respectifs du système RMH et de la liaison CAMAC vers le miniordinateur NORD : alors que le temps de lecture RMH ne demande que 150 ns par mot, l'interconnexion CAMAC exige 1,5 μ s pour transférer ce mot dans la mémoire NORD.

Lorsque la signature d'un événement est faite, le système d'acquisition est inhibé pendant tout le temps de transfert de l'événement. A faible taux de déclenchement, ce temps mort n'entraîne pas une très grande perte d'efficacité car la probabilité d'arrivée d'un bon événement pendant le transfert du précédent reste faible. Par contre dès que le taux de déclenchement augmente, l'efficacité du système chute rapidement. Le temps actif de l'expérience disparaît alors devant le temps d'acquisition et le coût extrêmement élevé d'une heure de faisceau rend inacceptable des efficacités de l'ordre de 50 %, c'est à dire plus de la moitié du temps faisceau inutilisé.

3.2. Limites liées à la structure informatique de l'expérience :

3.2.1. Le traitement des données par le système NORD :

Les informations des détecteurs sont compactées dans le RMH. Tous les événements sont transmis sans contrôle et sans discontinuité. Avant le transfert sur bande magnétique le miniordinateur NORD est donc obligé de redécouper et reformater l'information événement par événement, en ajoutant en début d'événement un entête qui le caractérise. Pour reconnaître un nouvel événement le programme doit décrire toutes les données et rechercher deux marqueurs de début d'événements, écrits par des modules encodeurs (CES). Ce traitement long, de par la lecture séquentielle de chaque donnée, limite d'autant les tâches d'histogrammation que pourrait exécuter l'ordinateur entre deux cycles d'acquisition de données. De plus, si pour une raison quelconque, ces indicateurs sont absents, toute l'information est inexploitable.

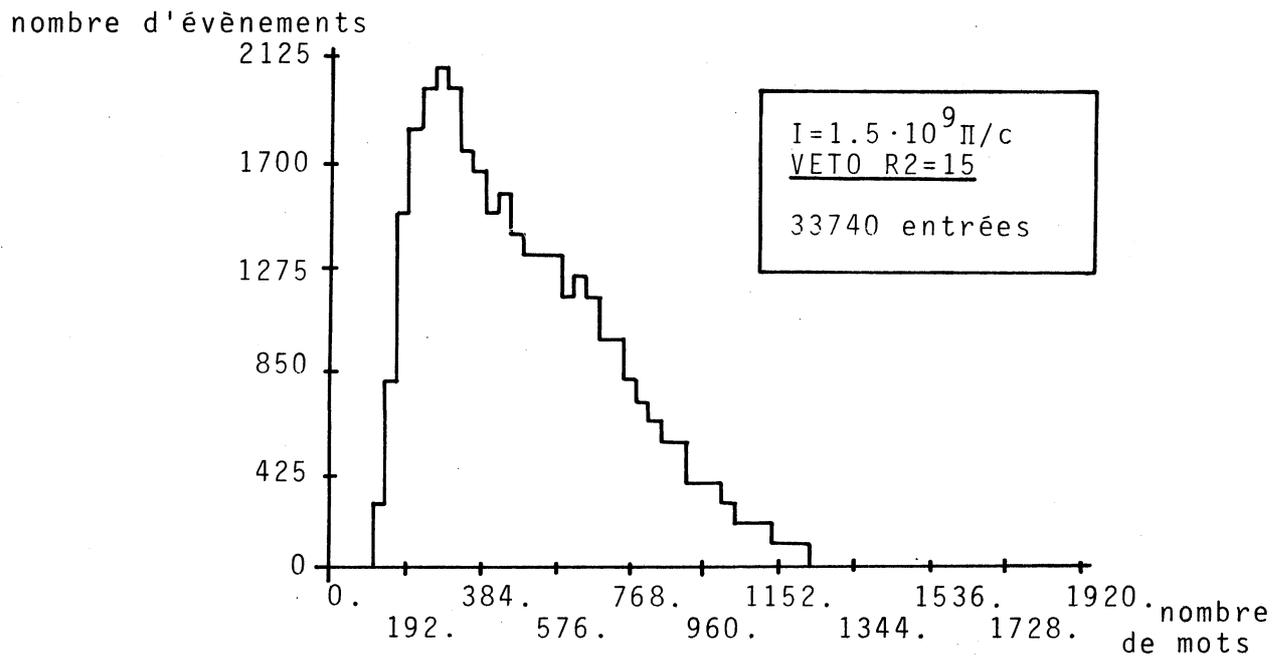


Figure 13.1. Histogramme du nombre total de mots par événement avec coupure multiplicité R

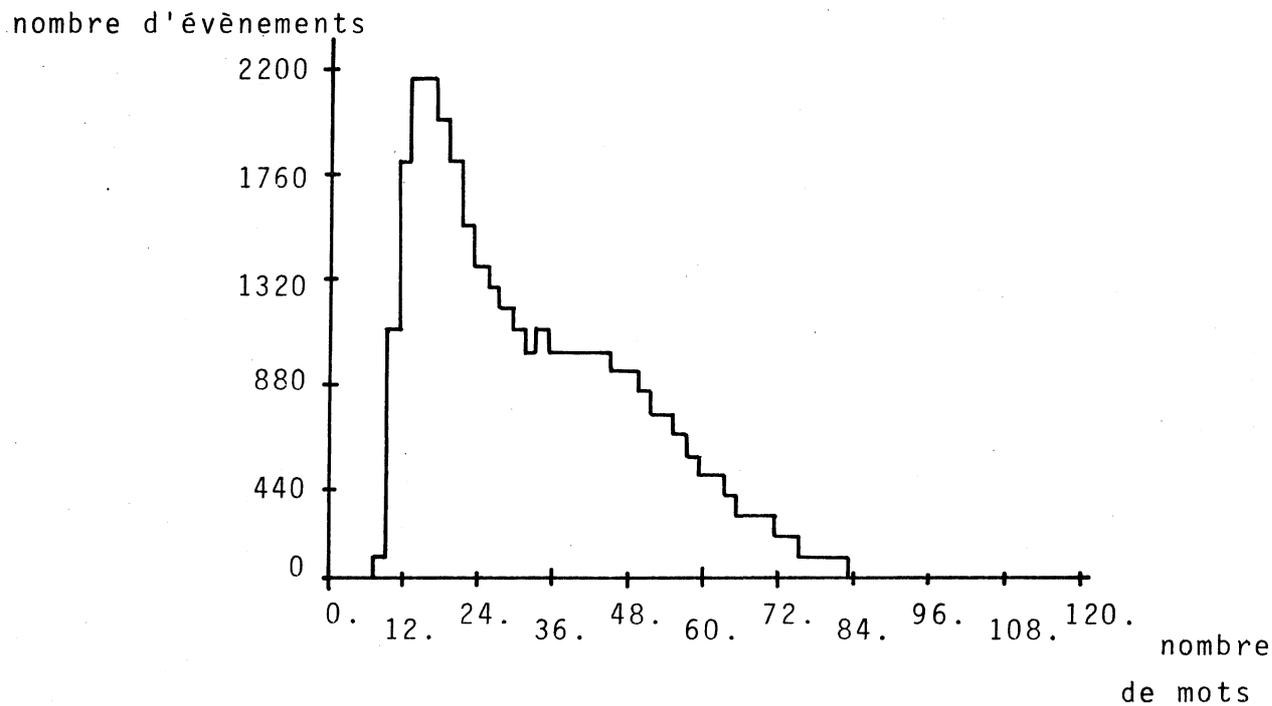


Figure 13.2. Histogramme du nombre de mots R et V par événement avec coupure multiplicité R

3.2.2 Le contrôle de l'expérience

Le miniordinateur ne peut contrôler la structure de l'information et sa validité qu'après l'avoir acquise et n'a pas accès au contrôle du transfert événement par événement.

3.3. Conclusion

L'analyse précédente montre la nécessité de placer dans la chaîne d'acquisition, un système capable d'harmoniser les flux entre la sortie du compacteur RMH et l'entrée de l'interconnexion CAMAC, afin d'améliorer le temps actif du système d'acquisition. Mais ce système doit également être capable:

- d'effectuer la gestion mémoire correspondante, car tout rejet implique un bouleversement de la mémoire d'accumulation. Une mémoire d'accumulation interne au système simplifie la programmation en évitant au processeur de filtrage, la gestion, toujours complexe, d'une mémoire externe (comme dans le cas de l'expérience WA2 (1975-78) ^[6] où la mémoire d'accumulation était la mémoire du miniordinateur NORD);
- de reformater et de prétraiter l'information événement par événement, avant transfert, de façon à libérer le miniordinateur pour les tâches d'histogramation et de tests.
- de surveiller le transfert de l'information et de contrôler la validité des données, afin de faciliter la détection des défauts de fonctionnement (absence de marqueurs, adresses compteurs non valides, ...).

III. LE SYSTEME DE FILTRAGE [7]

1. Architecture

L'amélioration du temps actif du système d'acquisition passe par la diminution du temps de blocage qu'entraîne le transfert des données vers le miniordinateur.

Pour réduire ce temps de blocage, il faut :

- réduire le nombre d'événements à transférer au miniordinateur en filtrant le flux des données,
- masquer les temps de transfert par la multiplication et le parallélisme des processeurs le long de la chaîne d'acquisition.

1.1. Réduction du flux des données

Pour réduire le nombre d'événements à transférer, on augmente la sélectivité du déclenchement en introduisant une sélection de second niveau. Deux types de sélection peuvent être appliqués à des instants et des endroits différents :

1. Le déclenchement logiciel :

Le traitement nécessaire à la sélection est effectué en parallèle avec le transfert des données, des mémoires associées aux détecteurs vers la mémoire d'accumulation. Si l'événement en cours de transfert ne satisfait pas aux conditions de l'algorithme, le transfert est interrompu et un nouvel événement peut être acquis. Cette sélection permet ainsi de réduire le temps mort introduit par le transfert de l'information vers la mémoire d'accumulation, mais nécessite :

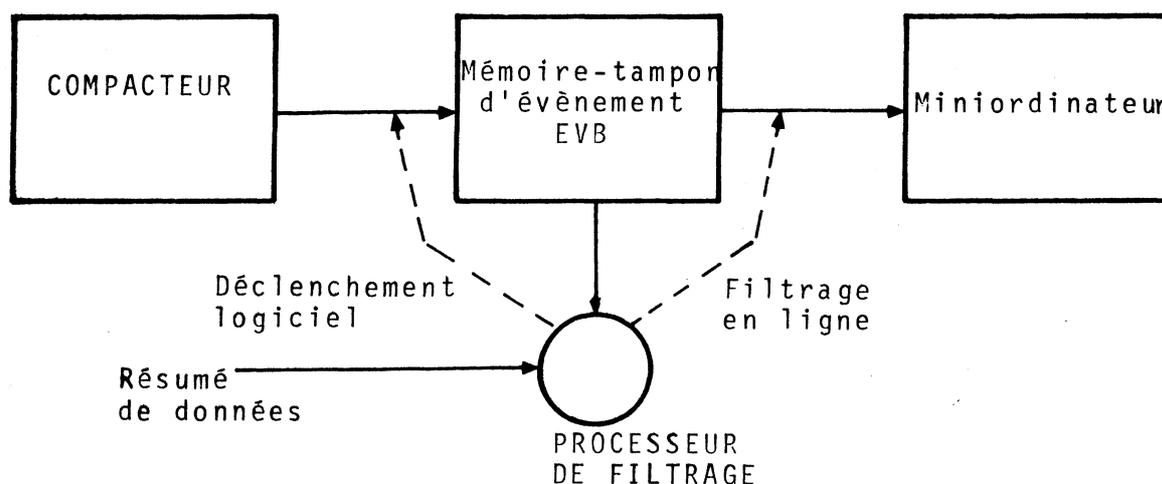
- un temps de décision inférieur au temps de transfert. C'est ici que la vitesse de traitement du processeur de filtrage est la plus critique car, pour être sélectif le déclenchement de second niveau nécessite des critères plus élaborés que ceux utilisés dans le déclenchement câblé de premier niveau et exige donc un algorithme de traitement plus complexe.
- de disposer très rapidement d'un résumé de données nécessaires à la décision, car la sélection ne peut se faire sur les données acquises.

2. Le filtrage en ligne :

La sélection s'effectue sur les données acquises, lorsque le transfert de l'information dans la mémoire d'accumulation est achevé.

La comparaison entre les temps de transfert RMH et CAMAC, respectivement 150 nsec/mot et 1,5 μ sec/mot, montre que pour masquer le temps mort introduit par le transfert CAMAC, le système de filtrage doit être capable d'acquérir les événements à la vitesse RMH et de réduire le flux de données d'un facteur 10. Une telle réjection difficile à obtenir à l'aide d'un seul traitement de sélection peut nécessiter plusieurs tests successifs, qui ne portent pas tous sur le résumé de données : la structure du système de filtrage doit permettre les deux types de sélection, déclenchement logiciel et filtrage en ligne.

Pour ce faire, on a recours à une mémorisation intermédiaire en plaçant entre le compacteur et la mémoire du miniordinateur, une mémoire-tampon compatible en entrée avec la vitesse de transfert RMH, en sortie avec la vitesse CAMAC, et accessible au processeur de filtrage.



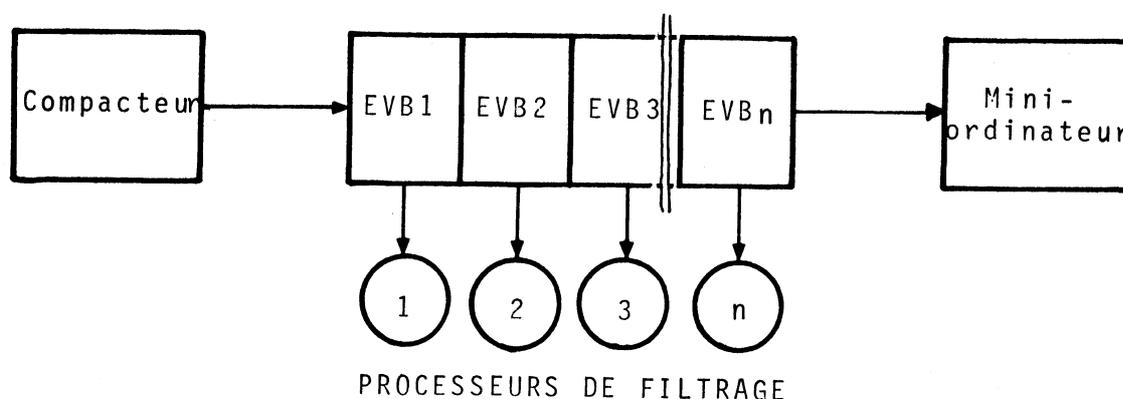
1.2. Masquage des temps morts

L'architecture décrite ci-dessus permet de réduire le temps mort global du système par la réduction du nombre de transferts vers le miniordinateur, mais n'évite pas le blocage du système d'acquisition pendant le transfert d'un bon candidat. Pour masquer ce temps mort, on remplace la mémoire-tampon EVB par un FI-FO constitué de plusieurs modules EVB parallèles, afin d'acquérir et de traiter de nouveaux événements pendant le transfert d'un événement précédent au miniordinateur.

Le temps d'exécution du traitement, appliqué comme déclenchement logiciel, peut varier avec la structure de l'événement à sélectionner, et demander pour certains événements plus de temps que le transfert de l'événement dans l'EVB. Pour que ce dépassement en temps n'introduise pas de temps mort supplémentaire dans le système, on associe à chaque module EVB son propre processeur de

filtrage : un nouveau cycle d'acquisition peut être lancé dès la fin du transfert de l'événement courant dans l'EVB, même si le traitement de cet événement n'est pas achevé.

Le nombre de couples EVB-processeur de filtrage est fixé par le temps d'exécution maximum du traitement, le temps limite avant introduction possible d'un temps mort étant égal au temps de transfert moyen que multiplie le nombre de couples EVB-Processeur de filtrage mis en parallèle.



1.3. Choix du processeur de filtrage

Aux impératifs de vitesse et de sélectivité, s'ajoute celui de la flexibilité du filtre. Tous les critères de filtrage ne peuvent être prévus et le traitement doit pouvoir s'adapter aux conditions nouvelles définies en cours d'expérience.

L'utilisation d'un filtre microprogrammé répond à ces contraintes, en réalisant un compromis entre le filtre câblé rapide mais figé, et le filtre programmé sur un miniordinateur, évolutif mais trop lent.

En outre, la conception et la réalisation d'une machine dédiée à l'expérience permet d'adapter au mieux sa structure au cahier des charges.

L'avantage d'un processeur microprogrammé dans la conception d'un système de filtrage réside en plusieurs points :

- la complexité du traitement est assurée en partie par le logiciel,

permettant une structure matérielle du noyau central assez générale et éventuellement évolutive. Cette banalisation permet de réduire le temps de conception et rend la maintenance plus facile;

- la réalisation d'entrées/sorties parfaitement adaptées à l'environnement permet un gain important des temps de lecture ou de transfert de données;
- Le nombre et la nature des interruptions nécessaires à la synchronisation du processeur avec l'expérience peuvent être définies pour répondre parfaitement aux conditions temporelles ou structurelles imposées par l'environnement.

La microprogrammation de l'ensemble du traitement est difficilement envisageable car la complexité de la microprogrammation rend difficile la modification rapide des algorithmes. Pour conserver la souplesse de programmation sans trop pénaliser la vitesse d'exécution, on a recours à un mode de programmation mixte:

- Le programme est une suite d'instructions microprogrammées;
- Chaque instruction est un microprogramme indépendant, appelé par un code opération qui fournit l'adresse de point d'entrée du microprogramme en mémoire de microprogramme.

Pour utiliser le calculateur en mode programmé, il faut donc disposer d'un jeu d'instructions complet, mais la capacité limitée de la mémoire de microprogramme (pour des raisons de vitesse) implique que seules les opérations fréquentes ou complexes soient microprogrammées. Le jeu d'instructions comprendra deux groupes d'instructions:

- les instructions générales nécessaires au fonctionnement courant (lecture/écriture mémoire et registres, opérations arithmétiques et logiques, débranchements), qui laisse la plus grande part de la capacité de la mémoire de microprogramme aux:
- instructions spécialisées, spécifiques à l'expérience, qui accroissent les performances du système de filtrage, par la microprogrammation complète des algorithmes longs ou complexes.

Cette programmation mixte permet d'écrire le programme de filtrage en Assembleur, ce qui confère, une grande facilité d'écriture et une grande souplesse de programmation.

2. Paramètres du filtrage [8]

2.1. Détermination du traitement

La simulation de l'expérience a permis de montrer qu'une coupure sur les valeurs de masse du dimuon approximées par la somme des impulsions transverses ($M = P_{T1} + P_{T2}$), introduit une perte des événements de masse supérieure à 5 GeV, d'environ 10 %. Pour réduire ce biais, il faut augmenter la précision sur la masse du dimuon en appliquant sa formule exacte :

$$M^2 = 2P_{i_1} P_{i_2} [1 - \cos\theta_1 \cos\theta_2 - \sin\theta_1 \sin\theta_2 \cos(\varphi_1 - \varphi_2)] \quad (1)$$

avec P_i l'impulsion du muon
 θ_i l'angle polaire
et φ_i l'angle azimutal

2.2. Précision du calcul de masse

La précision du calcul de masse effectué par le microprocesseur doit être de l'ordre du pour cent.

La formule de masse comporte des termes très petits du type $\sin\theta$ et des termes grands du type $\cos\theta$, $\cos \Delta\varphi$ et P_i . La simulation du calcul de masse sur grosse machine a permis de mettre en évidence l'insuffisante précision d'un calcul sur 16 bits. Pour atteindre la précision, des nombres flottants de 24 bits sont nécessaires.

2.3. Vitesse de traitement

Lorsque l'on travaille à flux incident élevé ($\sim 2 \cdot 10^9 \pi$), le taux de déclenchement moyen est de 1000 événements par seconde et la taille moyenne des événements, de 1000 mots (soit un temps de transfert moyen RMH-EVB de 100 μ s par événement). Pour pouvoir appliquer la coupure en masse comme déclenchement logiciel, le temps de traitement de cette coupure doit être inférieur au temps de transfert moyen RMH-EVB. Quand de nombreux sextants sont touchés, le temps de traitement de la coupure en masse augmente et peut être supérieur au temps de transfert RMH-EVB. Mais, pour éviter toute introduction de temps mort, ce temps de traitement doit rester inférieur au temps de transfert moyen RMH-EVB que multiplie le nombre de modules EVB mis en parallèle.

3. Choix des opérateurs câblés

3.1. Recherche du plus grand moment transverse

Pour ne pas perdre d'événements de hautes masses, le calcul de masse doit s'appliquer aux muons qui possèdent le plus grand moment transverse (P_T). La matrice de coïncidences délivrant 32 valeurs possibles par sextant, une recherche du plus grand P_T doit être effectuée dans chaque sextant avant le calcul de masse. 192 mots (32×6) doivent être traités, et pour chaque mot il faut :

- comparer de la valeur courante du P_T avec la plus grande valeur précédemment retenue
- sauvegarder cette valeur de P_T et le numéro du compteur V correspondant si la comparaison est positive (P_T courant $>$ P_T retenu)
- effectuer le comptage de multiplicité (nombre de compteurs V touchés par sextant)

En mode microprogrammé, le traitement de chaque mot nécessite 6 micro-instructions et le temps de recherche total est de 150 μ s. Pour réduire ce temps de traitement trop long par rapport au 100 μ s de temps de transfert RMH-EVB, la recherche du plus grand P_T est exécutée par un préprocesseur câblé qui fournit pour chaque sextant la valeur du plus grand P_T , le numéro du compteur V associé, la multiplicité des compteurs V et le numéro de sextant. A chaque sextant correspond un préprocesseur et les 6 recherches sont exécutées simultanément. Un tel ensemble permet de gagner un facteur 40 sur la vitesse de traitement en ramenant le temps d'exécution à 5 μ s.

3.2. Le multiplieur

Le calcul de masse du dimuon par la formule :

$$- \frac{M^2}{2} = P_{i_1} P_{i_2} (\cos\theta_1 \cos\theta_2 + \sin\theta_1 \sin\theta_2 \cos(\varphi_1 - \varphi_2) - 1)$$

demande 5 multiplications flottantes et 2 additions flottantes.

L'organigramme de la multiplication de 2 nombres flottants de 24 bits est donné fig.14. La multiplication des mantisses de 16 bits par une machine 24 bits nécessite un fractionnement selon l'arbre de Wallace.

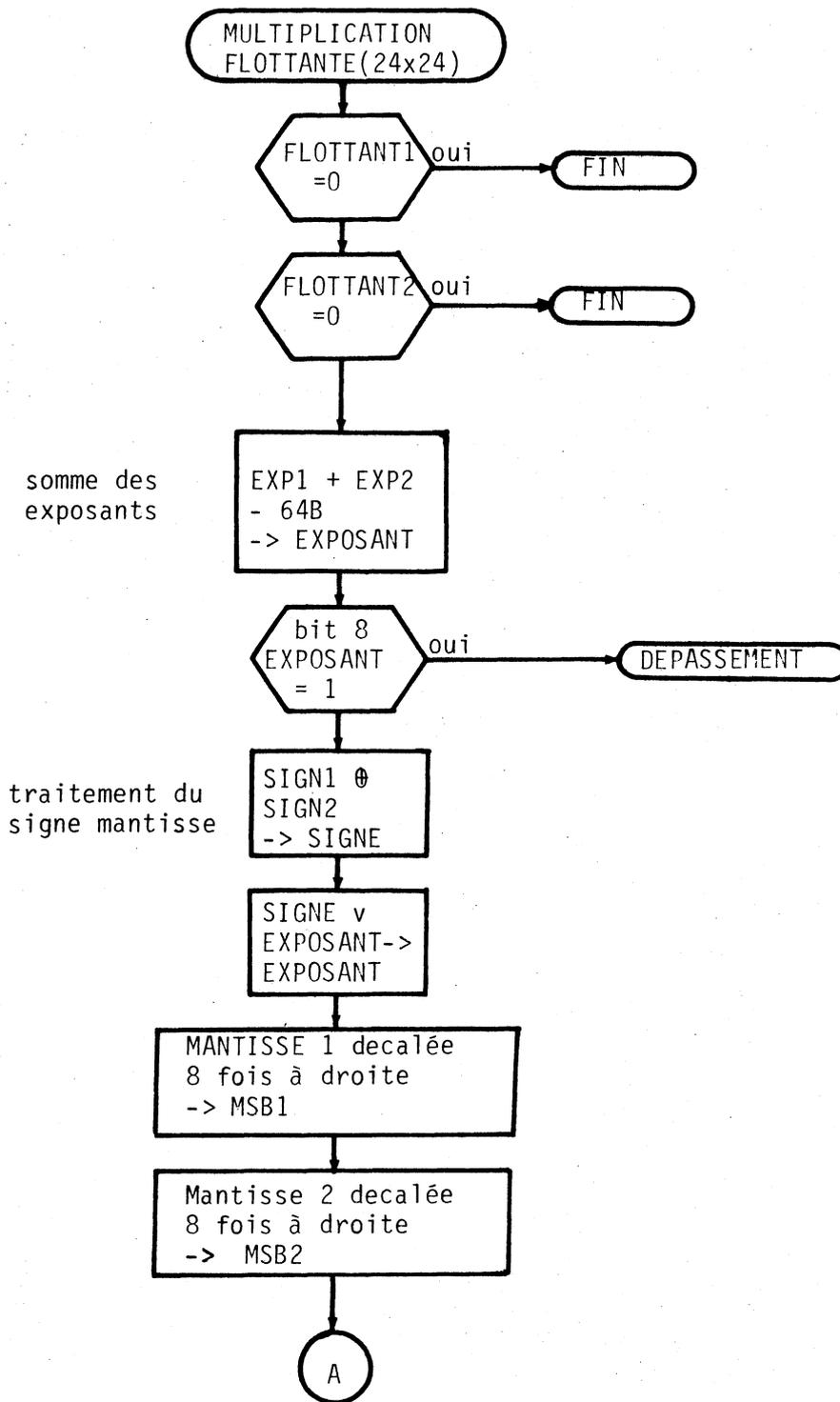
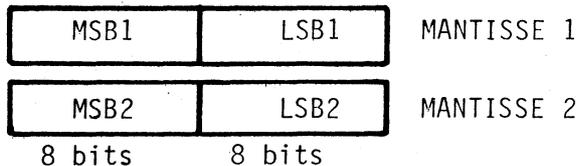
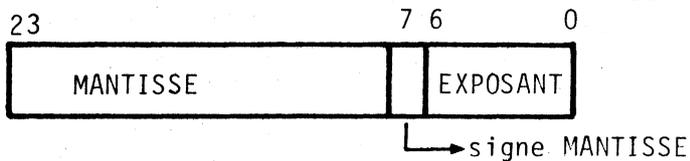
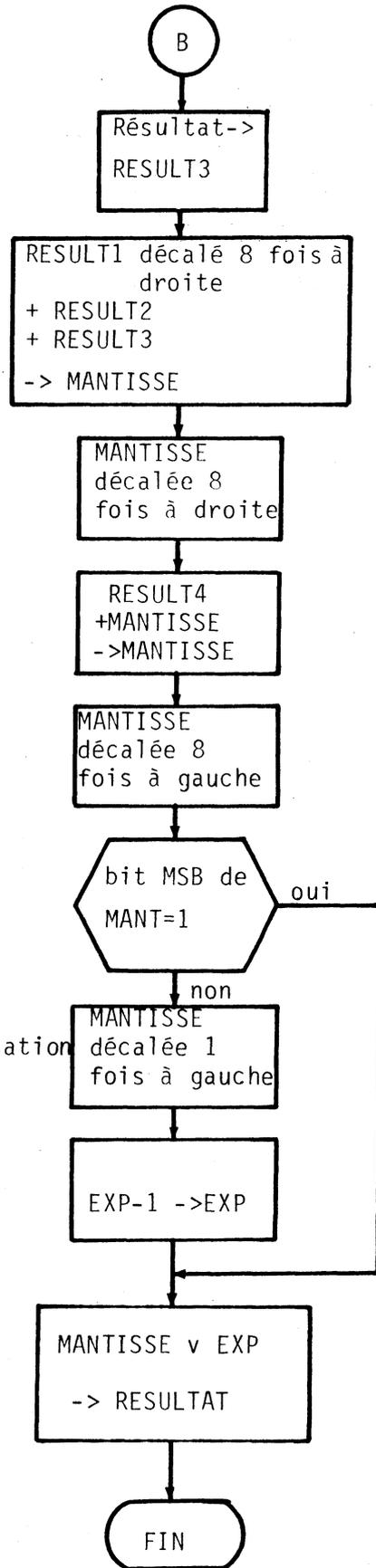
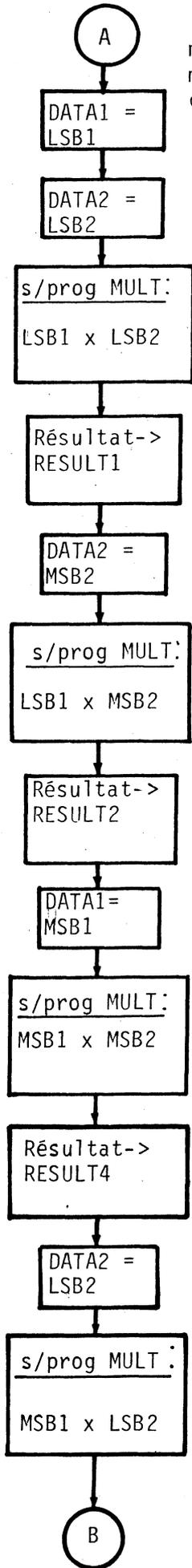
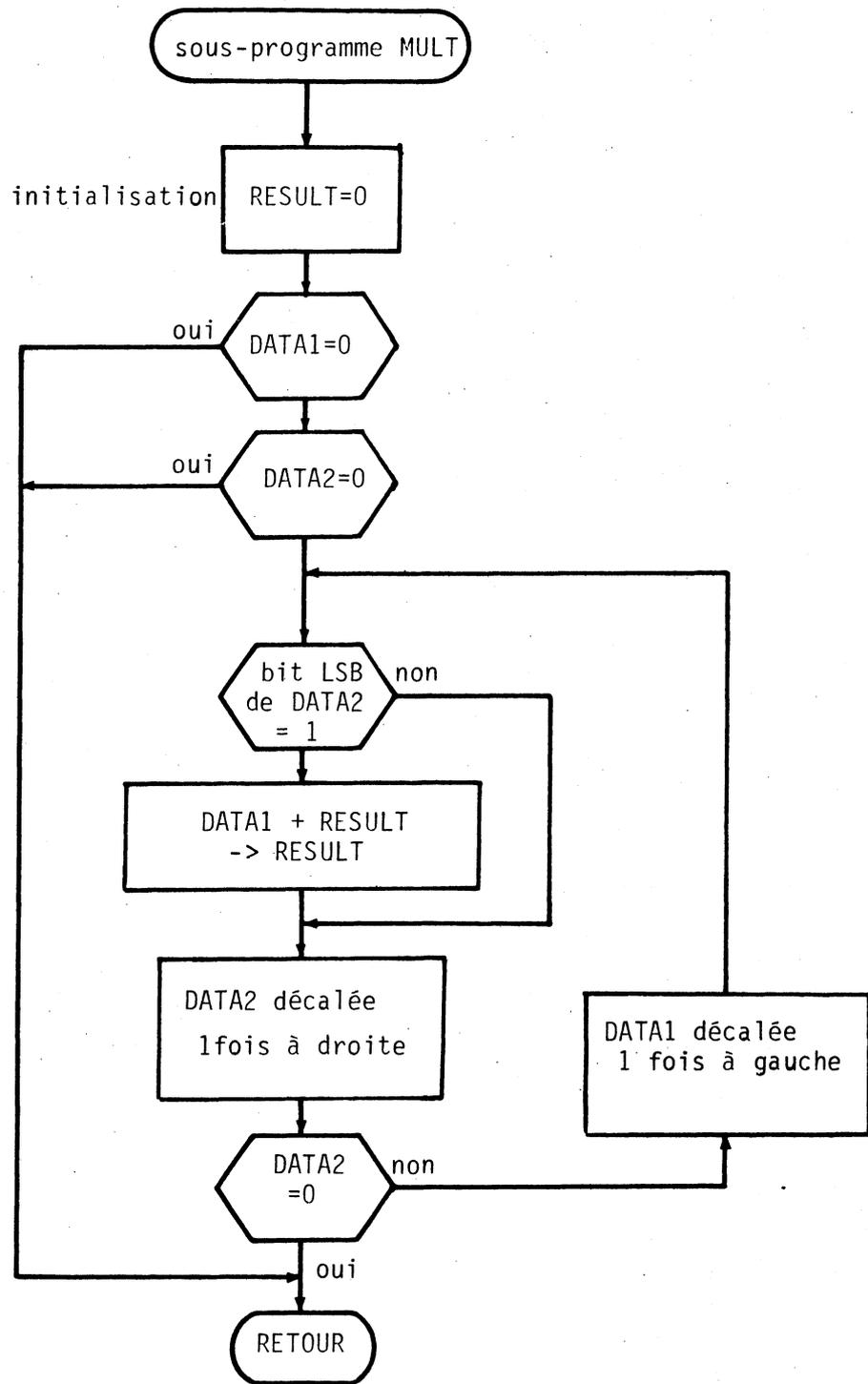


Figure 14 : Organigramme de la multiplication de 2 nombres flottants de 24 bits

multiplication des mantisses selon l'arbre de WALLACE



Renormalisation



En mode programmé GESPRO, cette multiplication demande 840 μ s (dans le cas le plus défavorable où tous les bits de mantisse sont à "1").

(Le programme Assembleur est donné en annexe A)

Pour réduire ce temps d'exécution prohibitif, la multiplication flottante peut être microprogrammée et s'exécute alors, en 51 μ s.

(Le microprogramme est donné en annexe A)

Le facteur 16 ainsi gagné sur le temps d'exécution n'étant pas suffisant, un opérateur arithmétique câblé a été développé. Conçu pour travailler, soit en entier, soit en flottant, il réalise les opérations de base suivantes:

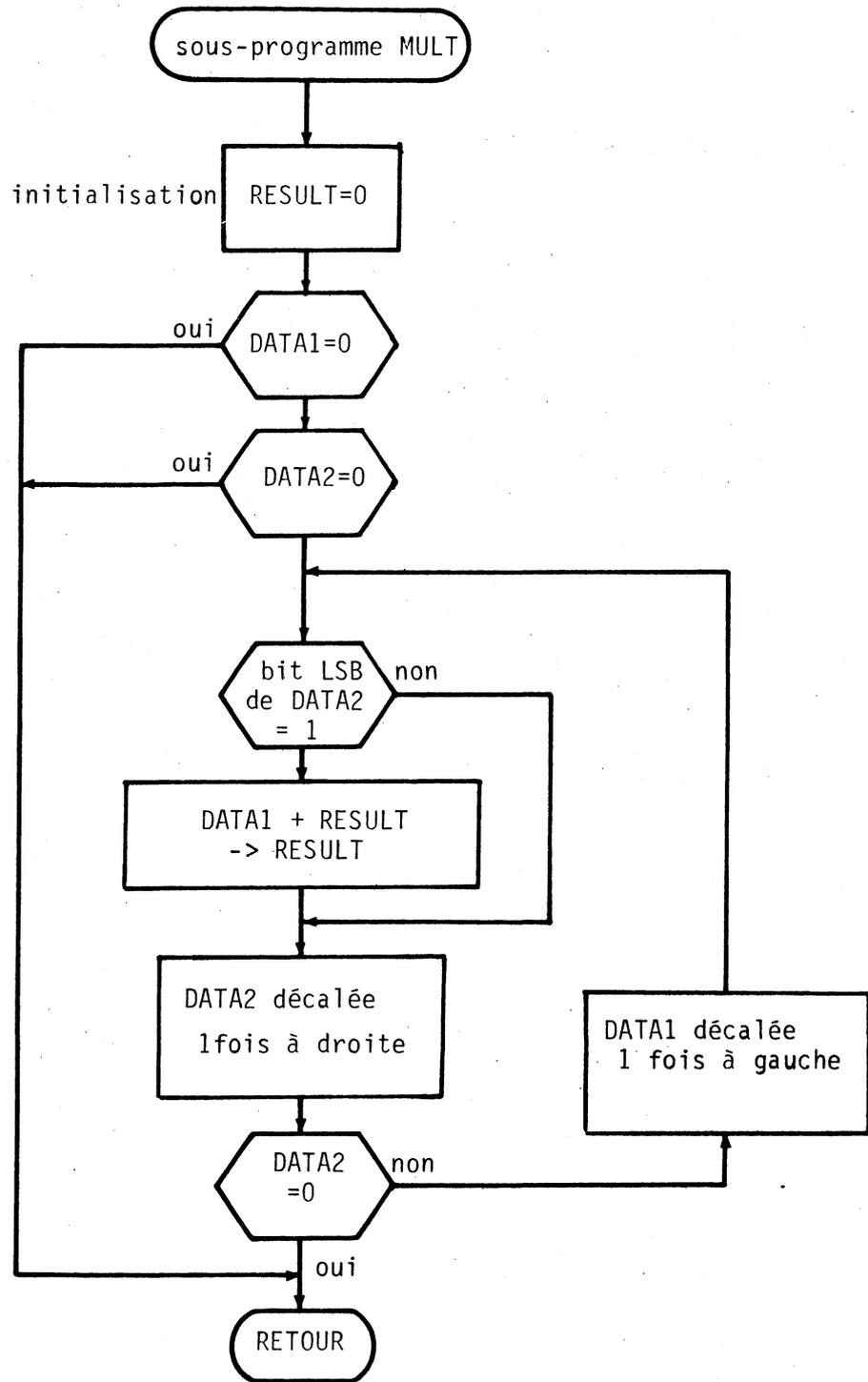
- multiplication entière (24 x 24 bits \rightarrow résultat 48 bits),
- multiplication flottante (24 x 24 bits \rightarrow résultat 24 bits),
- conversion Entier \rightarrow Flottant,
- conversion Flottant \rightarrow Entier,
- réduction au même exposant de 2 nombres flottants (nécessaire aux additions flottantes).

Toutes ces opérations sont effectuées en logique combinatoire, ce qui amène à des temps d'exécution de 200 ns. La multiplication est exécutée en 650 ns (chargement, exécution et lecture).

(Le bloc diagramme de la multiplication câblée est donnée en annexe A)

La carte multiplieur est banalisée, contrairement à la carte de recherche des plus grands P_T qui n'est dédiée qu'à l'expérience en cours. L'architecture et la structure des accès et des commandes de la carte multiplieur permettent de réutiliser facilement cet opérateur sur un autre calculateur et de rentabiliser ainsi son développement (Une carte multiplieur est associée au calculateur GESPRO développé pour l'expérience UA2).

Dans ces conditions, le temps de calcul de la masse d'un dimuon a été estimé à 25 μ s. Dans le cas le plus défavorable où les 6 sextants seraient touchés, 15 calculs de masse sont nécessaires pour un même évènement. La limite de 100 μ s par évènement impose donc une structure de 4 couples Unité Centrale - Multiplieur indépendants, capables de travailler en parallèle.



En mode programmé GESPRO, cette multiplication demande 840 μ s (dans le cas le plus défavorable où tous les bits de mantisse sont à "1").

(Le programme Assembleur est donné en annexe A)

Pour réduire ce temps d'exécution prohibitif, la multiplication flottante peut être microprogrammée et s'exécute alors, en 51 μ s.

(Le microprogramme est donné en annexe A)

Le facteur 16 ainsi gagné sur le temps d'exécution n'étant pas suffisant, un opérateur arithmétique câblé a été développé. Conçu pour travailler, soit en entier, soit en flottant, il réalise les opérations de base suivantes:

- multiplication entière (24 x 24 bits \rightarrow résultat 48 bits),
- multiplication flottante (24 x 24 bits \rightarrow résultat 24 bits),
- conversion Entier \rightarrow Flottant,
- conversion Flottant \rightarrow Entier,
- réduction au même exposant de 2 nombres flottants (nécessaire aux additions flottantes).

Toutes ces opérations sont effectuées en logique combinatoire, ce qui amène à des temps d'exécution de 200 ns. La multiplication est exécutée en 650 ns (chargement, exécution et lecture).

(Le bloc diagramme de la multiplication câblée est donnée en annexe A)

La carte multiplieur est banalisée, contrairement à la carte de recherche des plus grands P_T qui n'est dédiée qu'à l'expérience en cours. L'architecture et la structure des accès et des commandes de la carte multiplieur permettent de réutiliser facilement cet opérateur sur un autre calculateur et de rentabiliser ainsi son développement (Une carte multiplieur est associée au calculateur GESPRO développé pour l'expérience UA2).

Dans ces conditions, le temps de calcul de la masse d'un dimuon a été estimé à 25 μ s. Dans le cas le plus défavorable où les 6 sextants seraient touchés, 15 calculs de masse sont nécessaires pour un même événement. La limite de 100 μ s par événement impose donc une structure de 4 couples Unité Centrale - Multiplieur indépendants, capables de travailler en parallèle.

4. Le système d'acquisition avec EVB

4.1. Architecture

La figure 15 donne le schéma synoptique du système .

- Le système RMH est connecté aux modules EVB, gérés par un contrôleur câblé qui autorise le transfert des données du RMH vers les différents modules et leur transfert après traitement, vers la mémoire NORD à travers l'interconnexion CAMAC ;
- Chaque calculateur GESPRO est interconnecté à la mémoire multi-accès d'un module EVB et reçoit l'information des matrices de coïncidences à travers le préprocesseur câblé de recherche du plus grand moment transverse. A chaque unité centrale GESPRO est connecté un multiplieur, mais les cartes de recherche du plus grand P_T sont communes aux 4 GESPROS.

Le temps de transfert RMH-EVB est de 150 nsec/mot.

Le transfert CAMAC entre les modules EVB et la mémoire NORD est moins rapide que le transfert CAMAC RMH → NORD, car les modules EVB ne sont pas connectés sur le HIGHWAY du système CAMAC, mais sur une branche DATAWAY. Le temps de transfert EVB-NORD est de 2.5 µsec/mot.

4.2. Acquisition d'un événement

4.2.1. Principe

Lorsque le système de déclenchement reconnaît un bon candidat, la logique intersextants émet un signal d'acquisition qui échantillonne l'information des détecteurs dans les mémoires RMH et les P_T encodés dans les cartes de recherche des plus grands P_T .

Le contrôleur EVB activé par ce même signal, alloue un module EVB à l'acquisition des données RMH et active le calculateur GESPRO associé. A la fin du transfert, si un module EVB est libre, le contrôleur libère le système d'acquisition, qui est bloqué tant que l'information RMH n'est pas sauvegardée.

Pour autoriser le transfert des données vers la mémoire NORD, le contrôleur attend la réponse du calculateur activé.

4.2.2. Les signaux de contrôle :

Le diagramme des signaux de contrôle (fig. 16 et 17) visualise un cycle d'acquisition et les différentes branches de sortie.

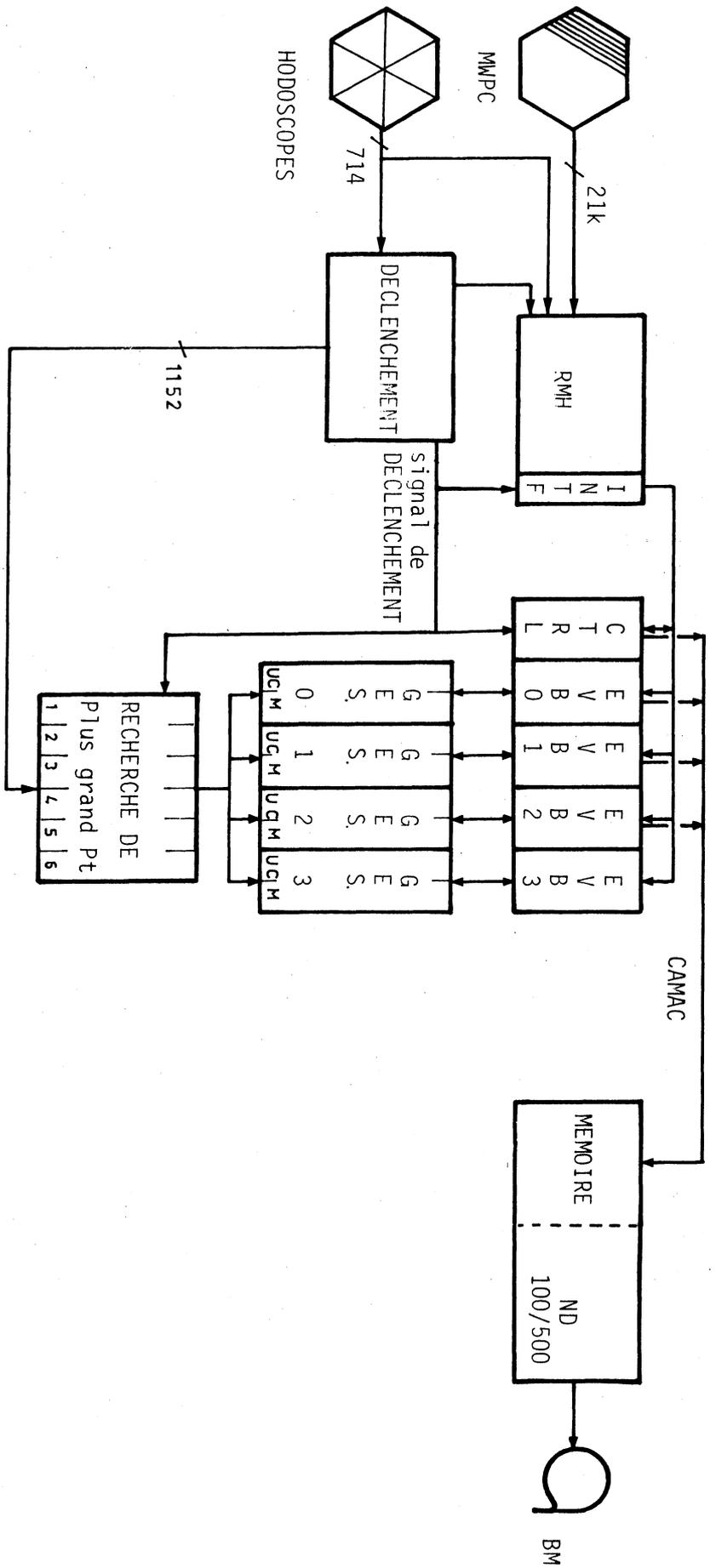


Figure 15 : Bloc diagramme du système d'acquisition avec EVB de l'expérience NA10

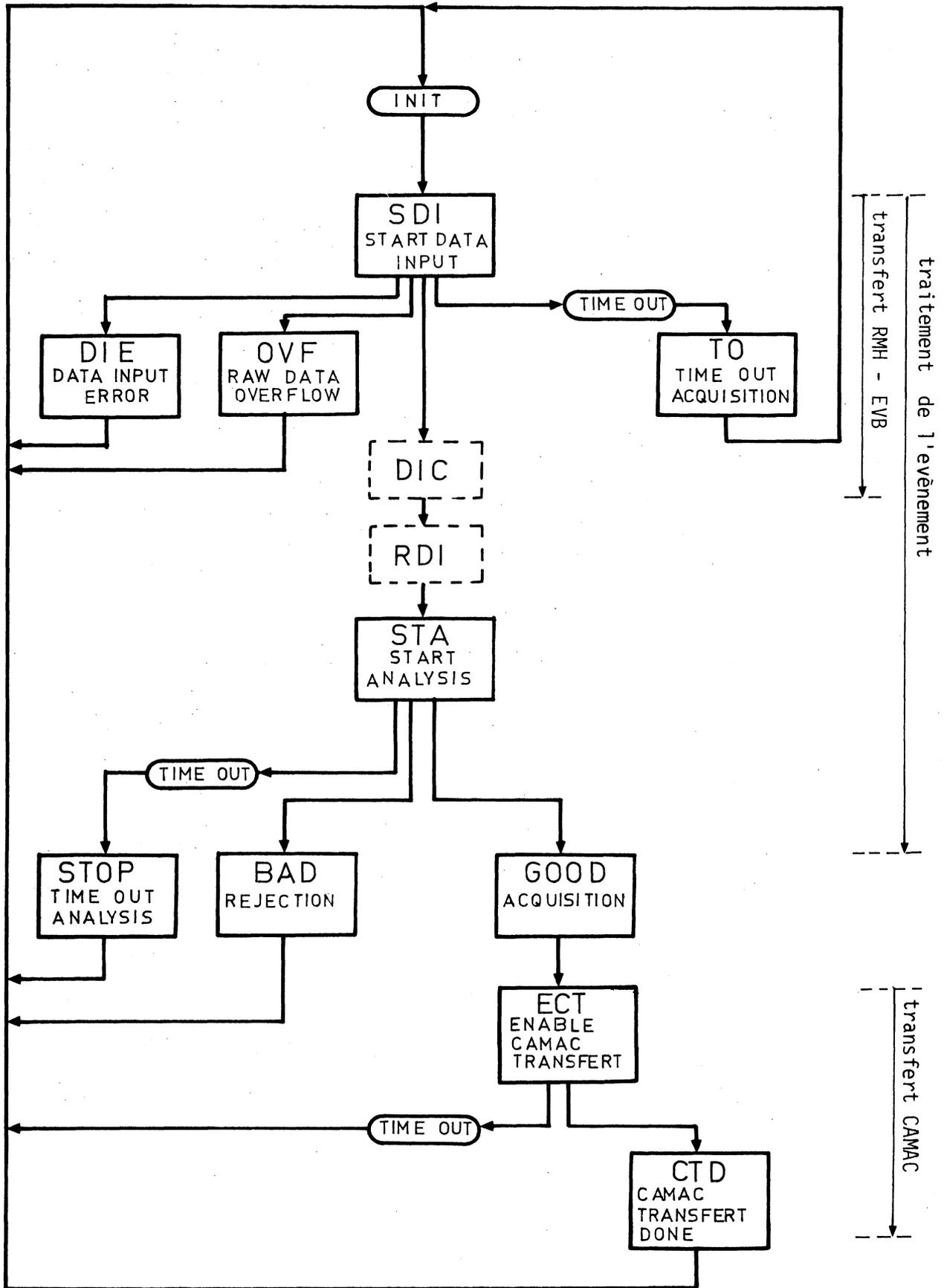


Figure 16 : Diagramme des signaux de contrôle EVB

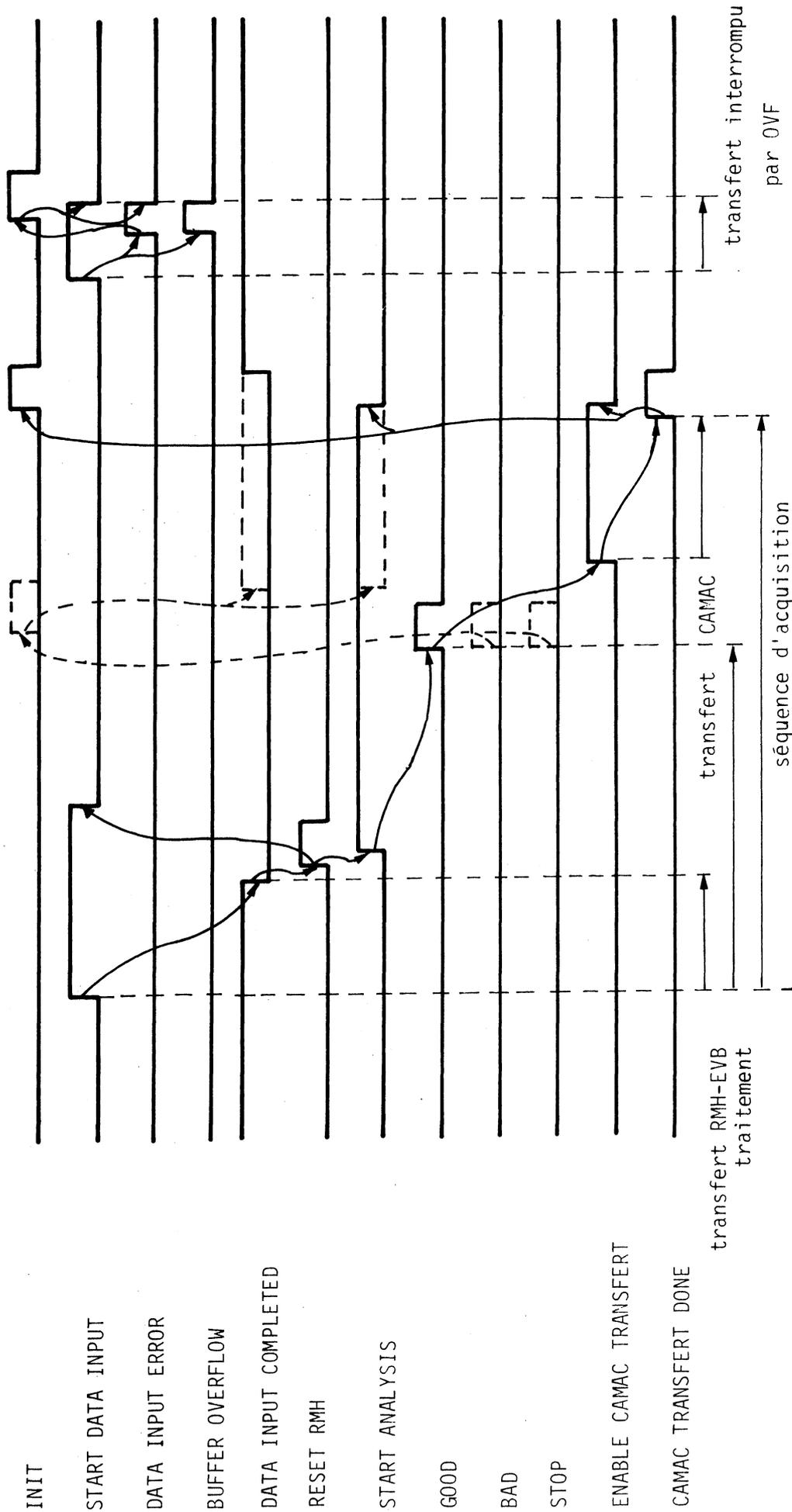


Figure 17 : Chromogramme des signaux de contrôle EVB

SDI (Start Data Input) est le signal de début d'acquisition d'un événement par l'EVB. DIE (Data Input Error) indique une interruption de transfert due à des données non valides. OVF (Overflow) est activé lorsque la taille de l'événement en cours de transfert est supérieure à la capacité de la mémoire EVB. La capacité physique de la mémoire est de 2K mots, mais cette limite est programmable pour permettre un rejet des événements longs, l'analyse ayant montré que les événements de plus de 1 K mots ne sont pas reconstituables et sont généralement dus au bruit de fond. Cette limite de la taille des événements est très efficace car elle permet d'augmenter le temps actif du système, en limitant le temps de transfert, directement lié à la taille des événements.

TO (Time Out Acquisition) indique que la fin du transfert signalé par DIC, (Data Input Completed), n'a pas eu lieu après SDI, dans un temps (programmable) suffisant pour transmettre les événements de taille maximale admise par le contrôleur. STA (Start Analysis) indique au calculateur, qu'il peut commencer l'analyse de l'événement sur les données acquises.

A partir de STA, le contrôleur attend une réponse du calculateur.

Si l'événement doit être conservé, GESPRO émet en fin de traitement signal GOOD et le contrôleur autorise le transfert CAMAC (Ensemble CAMAC Transfert). CTD (CAMAC Transfert Done), indique la fin du transfert et active le signal de réinitialisation INIT, qui libère le module pour un nouveau cycle d'acquisition.

En cas de rejet, GESPRO émet le signal BAD : le module est libéré pour l'acquisition d'un nouvel événement.

STOP est activé si après un intervalle de temps (programmable) après STA, le calculateur n'a émis aucune réponse.

4.3. La synchronisation GESPRO-système d'acquisition

4.3.1. L'interruption SDI

Lorsque le contrôleur EVB alloue un module EVB à l'acquisition des données RMH, le signal d'acquisition SDI active le programme de traitement du calculateur associé. L'analyse des chronogrammes d'acquisition montre que c'est à ce niveau que la vitesse d'exécution est la plus critique si l'on veut éviter toute introduction de temps mort, par le calculateur, dans le système d'acquisition.

4.3.2. Les registres d'état et de contrôle

Le registre d'état fournit au calculateur tous les signaux de contrôle du cycle d'acquisition émis par le contrôleur EVB. Ces signaux sont nécessaires à GESPRO, pour contrôler le status du transfert RMH-EVB, et lancer le programme de filtrage sur les données EVB, en fin de transfert.

La conception particulière du registre d'état (fig.18) permet de mémoriser toute activation des signaux de contrôle EVB. Tant que GESPRO n'a pas réinitialisé le registre, ces signaux sont présents côté GESPRO, même si le contrôleur les a désactivés.

L'ordre de rejet ou d'acquisition d'un événement (GOOD/BAD) analysé par GESPRO est transmis au contrôleur à travers le registre de contrôle.

4.3.3. Prise en compte du signal d'acquisition :

Les contraintes en temps, imposées à GESPRO se déterminent par l'analyse des signaux de contrôle dans le cas le plus défavorable :

1 seul GESPRO dans la chaîne d'acquisition et la séquence :

SDI : GESPRO démarre un cycle d'acquisition



DIE : le transfert RMH-EVB est interrompu par une erreur de transfert
↓ sur la lère donnée

SDI : GESPRO est aussitôt réalloué à un nouvel événement

Pour chaque événement GESPRO doit effectuer :

- 1 comptage de cet événement
- 1 contrôle du registre d'état pour vérifier de quelle manière s'est achevé le traitement précédent et, le cas échéant, comptabiliser les sorties en erreur ou rejet (DIE, OVF, TO, STOP). La lecture du registre doit être faite avant que le transfert en cours puisse modifier son contenu par l'activation d'un signal d'interruption de transfert.

Les cartes de P_T étant communes aux 4 calculateurs, aucun nouvel événement peut être acquis avant le transfert des mots P_T de l'événement courant dans la mémoire du calculateur correspondant.

Le temps de traitement, nécessaire à l'exécution de ces tâches, fixe l'intervalle de temps limite qui doit séparer deux déclenchements. Ce temps doit être le plus court possible pour ne pas introduire un temps mort trop important dans le système d'acquisition. L'intervalle entre 2 SDI a été fixé, par le contrôleur EVB, à 15 μ sec.

STATUS EVB

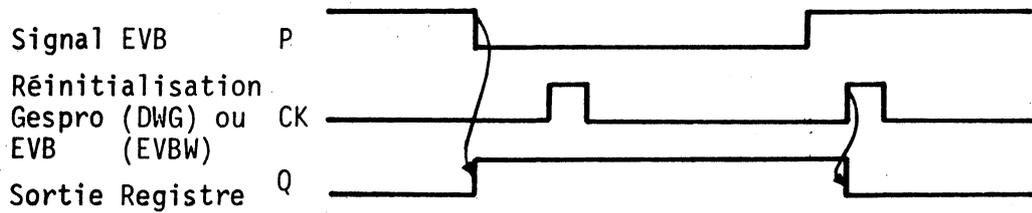
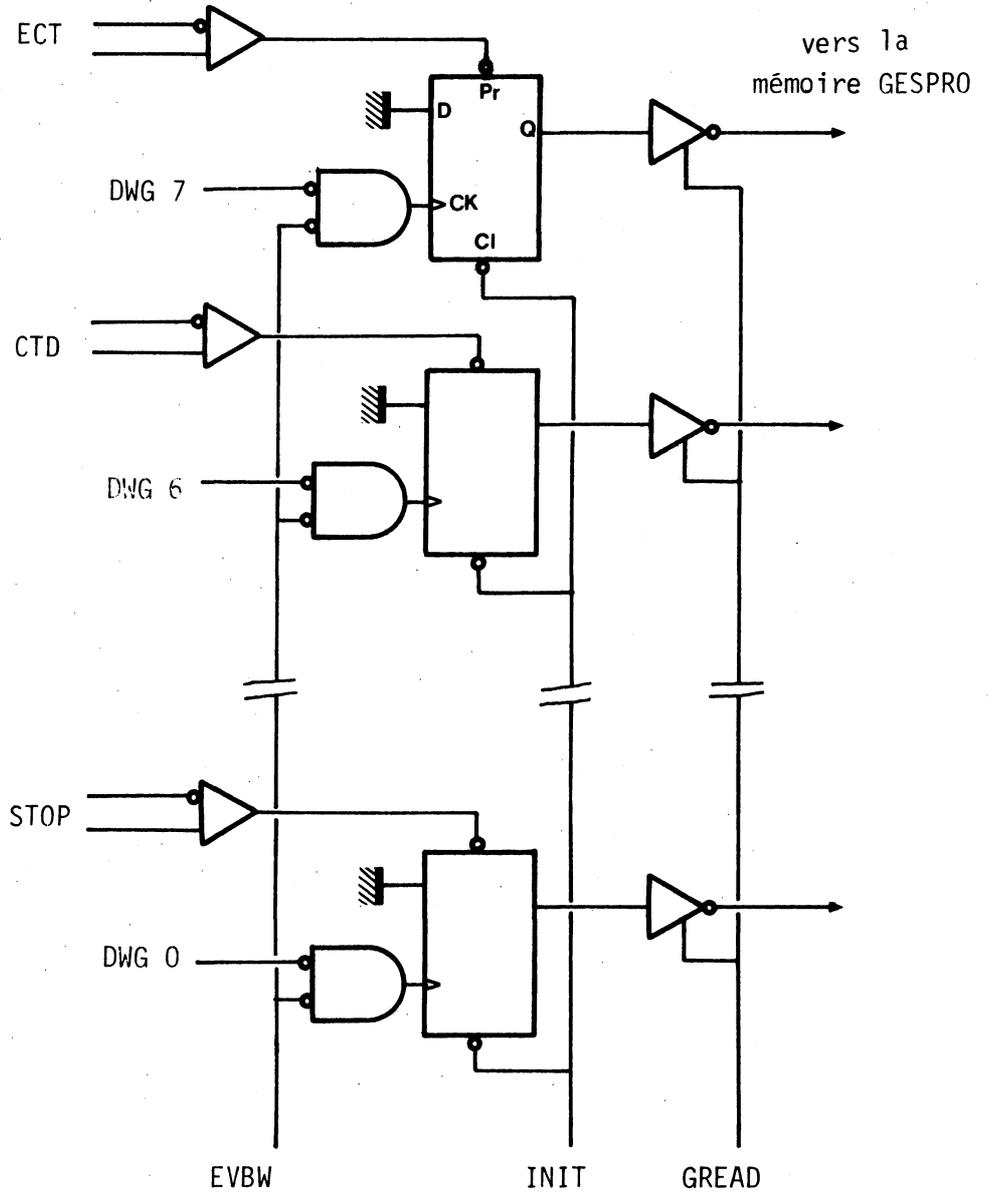
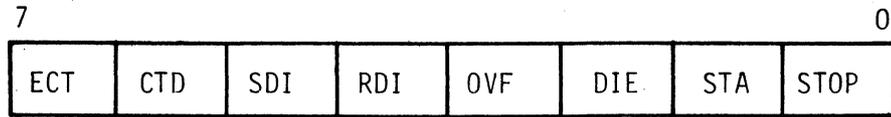


Figure 18 : Structure du registre d'état

Le signal SDI peut être pris en compte de trois facons:

1. Le registre d'état est testé à intervalles réguliers jusqu'à l'apparition du signal.

Cette méthode n'est envisageable que lorsque le calculateur attend le signal d'acquisition, à la fin du programme de filtrage de l'évènement courant. Quand le cycle d'acquisition est interrompu par le contrôleur EVB et le module EVB réalloué à un nouvel évènement, GESPRO n'est pas en attente, et dans ce cas, le temps de réponse du calculateur est prohibitif. Il peut varier entre 2 μ s et plusieurs dizaines de μ s, selon l'instant d'activation du signal par rapport aux instructions de test SDI, réparties dans le programme d'acquisition.

2. Le signal est appliqué comme une interruption externe classique.

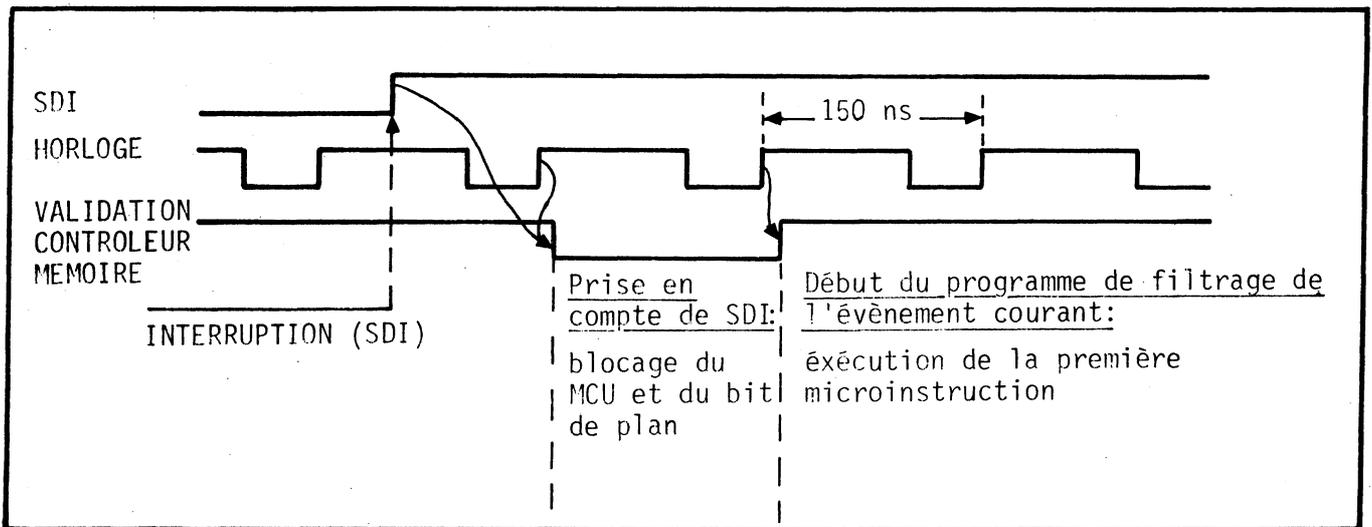
On évite, ainsi, de tester inutilement le registre d'état, mais dans les conditions définies précédemment (15 μ s entre 2 SDI), cette méthode ne peut être retenue car la prise en compte de l'interruption n'est pas assez rapide:

- GESPRO ne prend en compte les interruptions que pendant la phase de recherche de l'instruction suivante. Plusieurs μ s risquent de s'écouler entre l'instant d'activation de l'interruption et sa prise en compte, si GESPRO exécute une instruction longue lorsque SDI est activé;
- la prise en compte d'une interruption classique implique un changement de contexte avec sauvegarde qui entraîne un délai de 4 μ s entre la prise en compte de l'interruption et le début d'exécution du programme associé.

3. Le signal est appliqué directement au contrôleur de la mémoire de microprogramme et bloque le déroulement normal du microprogramme en cours.

L'interruption SDI est prise en compte au cycle horloge suivant l'instant l'instant d'activation du signal.

Moins de 300 ns après l'interruption, le calculateur commence à exécuter le programme de filtrage de l'évènement en cours d'acquisition, comme le montre le chronogramme qui suit:



Le changement de séquence s'effectuant sans sauvegarde et sans changement de contexte, certaines précautions de programmation doivent être prises, notamment au niveau du compteur ordinal et de la gestion des interruptions.

5. Le calculateur GESPRO

GESPRO est une machine 24 bits, microprogrammable sur 48 bits de 150 ns de temps de cycle. Son architecture est représentée figure 19.

5.1. Architecture

5.1.1. L'unité centrale

Réalisée autour des circuits INTEL de la famille 3000, elle se compose:

- d'une unité arithmétique et logique de 24 bits, constituée par la mise en parallèle de 12 circuits CPE INTEL 3002^[9], microprocesseurs en tranche de 2 bits. Sa structure multi-busses (2 busses de sortie A et D, 3 busses d'entrée M, I et K) lui permet un grand parallélisme de traitement: deux périphériques qui n'utilisent pas les mêmes busses peuvent être activés en simultanéité. (Les fonctions CPE sont donnés en annexe B.)
- d'une unité de contrôle, le MCU INTEL 3001^[9], qui gère la mémoire de microprogramme. A partir d'un code de saut, contenu dans la microinstruction courante et de son adresse, il calcule l'adresse de la microinstruction suivante. La logique du calcul peut tenir compte, selon la nature du saut utilisé, de la valeurs des indicateurs d'états ou du code appliqué sur deux busses d'entrée (Sx, Px) en provenance du bus mémoire. Le MCU permet également de prendre en compte les interruptions sur un code de saut particulier et d'effectuer un débranchement câblé. La capacité d'adressage est de 512 microinstructions, organisées dans une représentation matricielle de 32 lignes x 16 colonnes. (Les codes de saut sont donnés en annexe B.)
- d'une mémoire vive de microprogramme organisée en 2 plans de 512 mots de 48 bits. Cette séparation en plans, est invisible au contrôleur MCU et nécessite l'adjonction d'un bit opérateur qui complète l'adresse [ligne, colonne] par un numéro de plan. Un accès CAMAC permet le chargement initial des microprogrammes mais il est bloqué dès que le processeur fonctionne.

Toutes les liaisons entre l'Unité Centrale et les périphériques sont dotées de registres-tampons, de façon à masquer les temps d'accès par recouvrement: toute l'information nécessaire à un périphérique est mémorisée dans ses registres tampons lors de son activation, afin de libérer l'Unité Centrale qui peut exécuter des tâches qui n'affectent pas ces registres pendant le temps d'accès au périphérique (modification du bus Adresse pendant une lecture mémoire, ...).

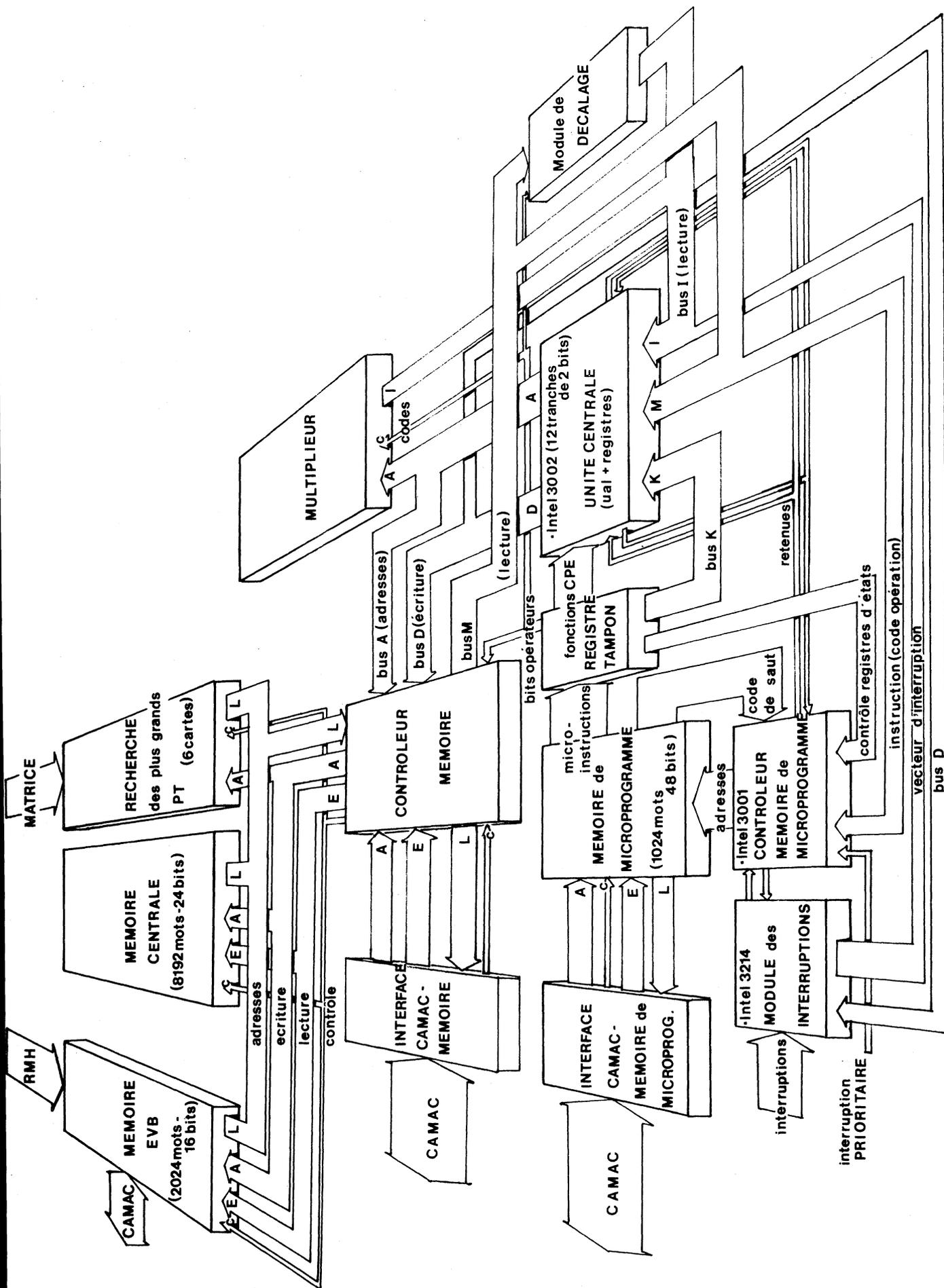
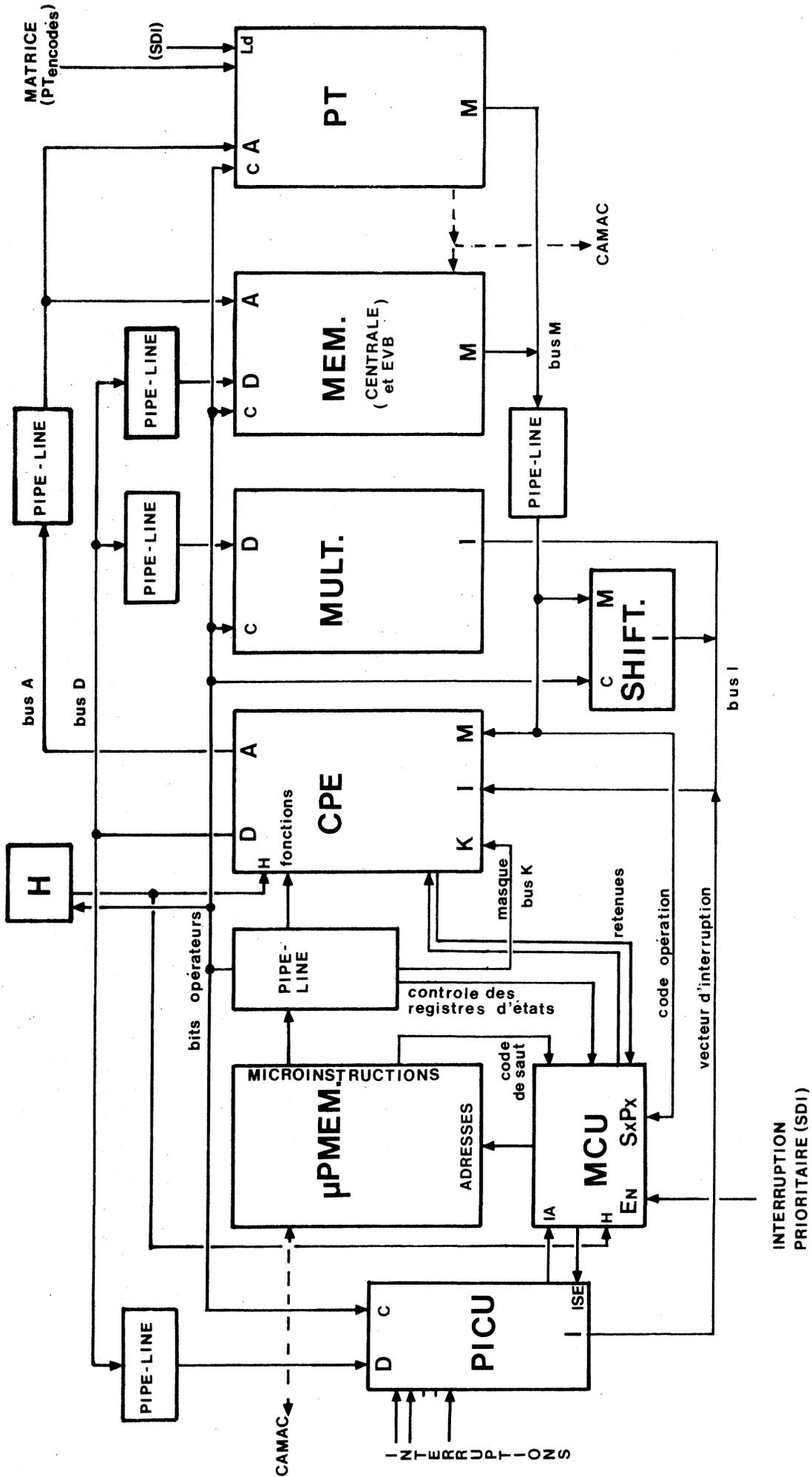


Figure 19: Architecture du calculateur GESPRO



- Architecture du calculateur GESPRO -

Un bit opérateur (READ) permet à l'unité centrale de se resynchroniser avec les périphériques. Lorsqu'un périphérique est activé, il émet un signal d'occupation BUSY pendant toute la durée du cycle d'occupation. Si la demande de resynchronisation READ se présente alors que le signal d'occupation est actif, l'horloge monophasé qui séquence l'unité centrale s'arrête, et ne redémarre qu'à la disparition de ce signal (fig. 20).

Un registre-tampon, placé entre la sortie de la mémoire de microprogramme et l'unité arithmétique et logique, mémorise le contenu de la microinstruction courante, excepté le champ Adresse, directement appliqué au MCU. Pendant l'exécution de l'opération sur l'unité centrale, le MCU calcule l'adresse et recherche la microinstruction suivante. Cette recherche en recouvrement permet de masquer le temps d'accès à la mémoire de microprogramme, mais introduit un délai d'une microséquence entre le fonctionnement du MCU et du CPE, qui exige une grande attention de microprogrammation quant à l'exécution des sauts conditionnés par le résultat d'une opération CPE.

Les commandes de démarrage et d'arrêt de l'horloge sont transmises à GESPRO par des ordres CAMAC décodés dans l'interface CAMAC-Unité Centrale.

5.1.2. Les périphériques

a) Le module d'interruptions

Un module d'interruptions, construit autour d'un Encodeur de Priorité INTEL 3214^[9], gère huit niveaux d'interruptions internes ou externes.

b) L'opérateur de décalage

Pour exécuter rapidement les opérations de décalage multiple, un opérateur câblé entre le bus mémoire M et le bus I, permet une rotation en 3 groupes de 8 bits.

c) Le multiplieur

3 bits de code définissent une fonction par les 8 fonctions nécessaires au fonctionnement des 5 opérations de base (fig. 21). Les opérandes (bus D) sont mémorisés lors de l'appel multiplieur (ENX CODE W). Le bus D est alors libéré, mais comme l'opérateur est entièrement combinatoire, le code doit rester appliqué jusqu'à la lecture du résultat (bus I).

Un indicateur de dépassement de capacité (OVF) est disponible sur une ligne séparée.

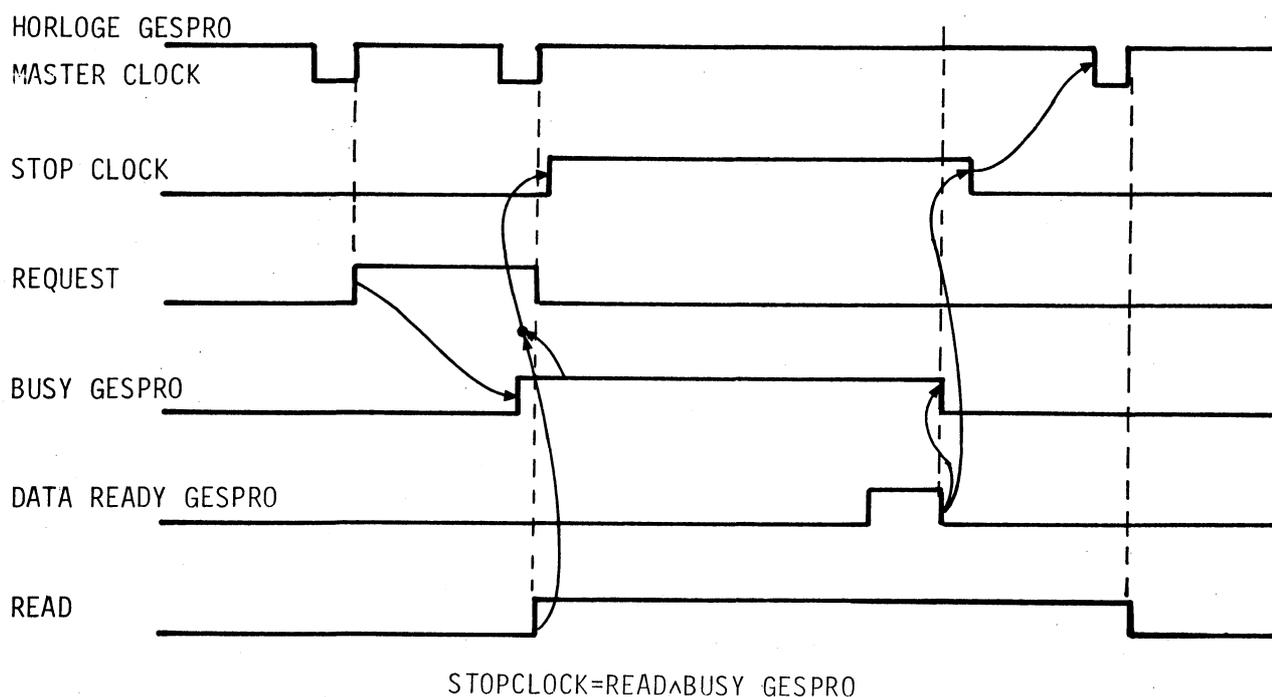


Figure 20 : Chronogramme de la resynchronisation unité centrale - périphériques

C2	C1	C0	ENX	OPERATIONS
0	0	0	1	Réduction au même exposant: chargement 1er opérande (W=1) / lecture 1ere donnée réduite
0	0	1	1	Réduction au même exposant: chargement 2e opérande (W=1) / lecture 2e donnée réduite
0	1	0	1	Multiplication flottante: chargement 1er opérande (W=1) / lecture du résultat
0	1	1	1	Multiplication flottante: chargement 2e opérande (W=1)
1	0	0	1	Multiplication entière: chargement 1er opérande (W=1) / lecture du résultat MSB
1	0	1	1	Multiplication entière: chargement 2e opérande (W=1) / lecture du résultat LSB
1	1	0	1	Conversion flottant->entier: chargement opérande (W=1) / lecture du résultat
1	1	1	1	Conversion entier->flottant: chargement opérande (W=1) / lecture du résultat

Figure 21 : Codes multiplieur

La carte "multiplieur" est conçue en TTLS et utilise des multiplieurs 12 bits.

Du chargement des données à la lecture du résultat, les multiplications entière et flottante, les conversions et la réduction au même exposant demandent 200 nsec.

d) La mémoire centrale

GESPRO dispose d'une mémoire centrale de 8 K mots de 24 bits à laquelle s'ajoutent 2 K mots de 16 bits de la mémoire EVB. La mémoire est accessible par CAMAC à travers un interface et une gestion d'accès permet l'écriture ou la lecture mémoire en vol de cycle. Un cycle GESPRO demande 400 nsec, mais toutes les liaisons avec l'unité centrale sont dotées de registres-tampons, autorisant la lecture et l'écriture mémoire en recouvrement.

Lorsqu'un appel GESPRO à la mémoire EVB n'est pas pris en compte au bout de 400 μ sec, un signal de dépassement (Time OUT EVB) est émis et peut être traité comme une interruption externe.

e) Les cartes de recherche du plus grand P_T :

Ces cartes sont adressées comme la mémoire (bus A), chaque carte ayant son adresse propre.

Les P_T encodés, fournis par la matrice sont mémorisés dans les registres d'entrées par le signal d'acquisition.

5.2. Format des microinstructions

Le format général d'une microinstruction GESPRO est donné figure 22.

- . Les 17 bits de poids faible sont les bits du bus de masque K.
- . Les 7 bits suivants (F_0 à F_6) définissent les fonctions de l'unité arithmétique et logique, spécifiées par la notice INTEL : F_4 , F_5 et F_6 définissent le groupe F, et, F_0 à F_3 le numéro de registre.
- . Les bits 24 à 35 décrivent le champ des bits opérateurs propres à la structure du calculateur. Ils permettent l'adressage et le contrôle des opérateurs câblés qui ont été ajoutés au noyau de base pour augmenter les performances du système :

- le bit IT autorise un débranchement conditionné par un bit particulier du bus mémoire en cas d'adressage indirect ou indexé, ou par l'indicateur de dépassement de capacité (OVF) du multiplieur. Ce débranchement est invisible au MCU.
- les deux bits MI1 et MI2 sont des modificateurs d'instructions: de nombreuses instructions générales ne diffèrent que par le numéro de registre utilisé; pour ne pas écrire autant de microinstructions qu'il y a de registres, les modificateurs remplacent les deux bits F0 et F1 du champ CPE de la microinstruction:
 - soit par les deux bits de poids faible du code opération (Modificateur MI2),
 - soit par les bits 2 et 3 du code opération (Modificateur MI1).Pour les 4 instructions, le microprogramme est le même et c'est le code opération qui change en fonction du registre sélectionné.
- les bits REQ, READ, W, ENX, $C_2C_1C_0$, sont les signaux de contrôle et de synchronisation des périphériques:
 - REQ lance un cycle mémoire: d'écriture si W = "1", de lecture si W = "0";
 - READ est de bit de synchronisation des périphériques avec l'Unité Centrale;
 - les 3 bits de code $C_2C_1C_0$ sont alloués au multiplieur si ENX = "1", aux périphériques du noyau si ENX = "0": module des interruptions ou module de décalage;
- le bit P est un bit d'adresse supplémentaire qui permet de sélectionner le numéro de plan de la mémoire de microprogramme.

Tous ces bits sont indépendants et activables simultanément. Cette simultanéité peut être illustrée par l'exemple suivant:

Conversion Entier → Flottant d'une donnée mémoire

ADRESSE →	MAR Req	Lancement du cycle lecture mémoire
M →	AC Read	Attente de la fin du cycle et lecture de la donnée
I →	AC ENX W $C_2C_1C_0$	Read ReqChargement de la donnée dans le multiplieur
			.Lancement du cycle multiplieur pour la fonction $C_2C_1C_0$
			.Attente de la fin du cycle multiplieur
			.Lecture du résultat de la conversion
			.Lancement du cycle d'écriture du résultat en mémoire.

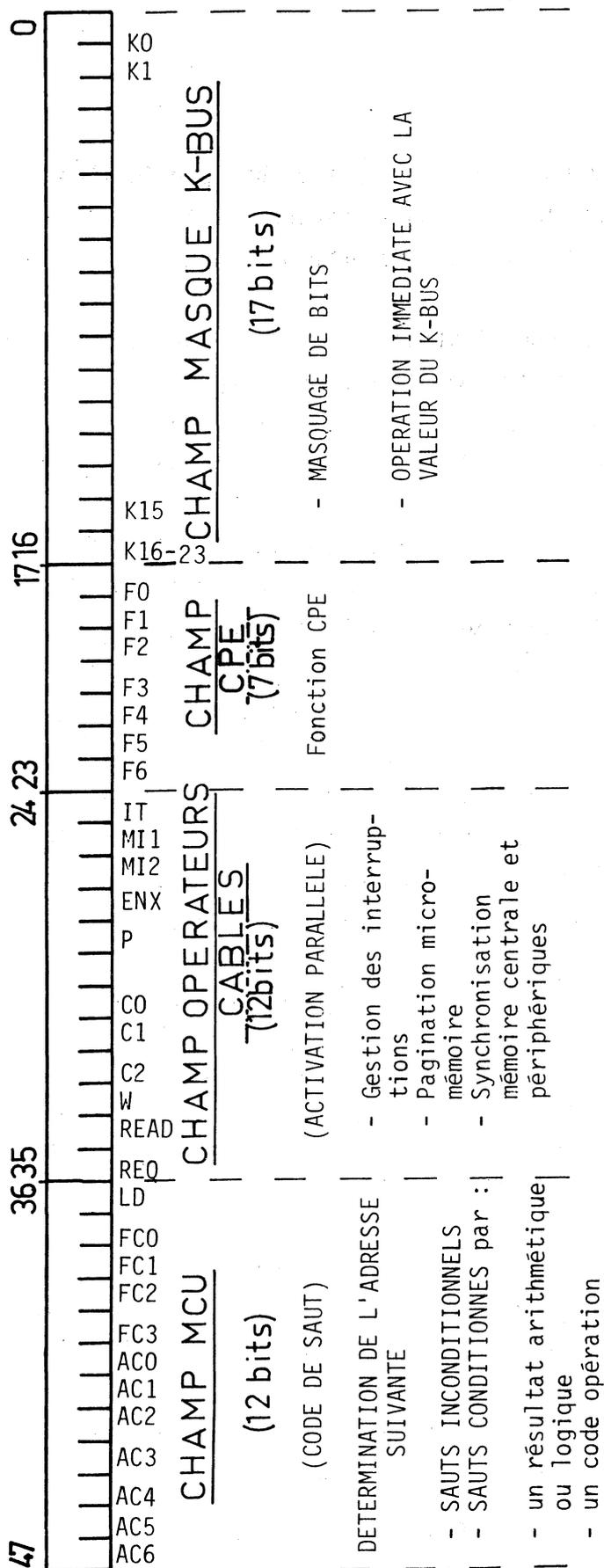


Figure 22 : FORMAT DES MICROINSTRUCTIONS GESP RO

Cette séquence s'exécute en 960 ns. L'équivalent programmé avec le multiplieur cablé demande 4 μ s:

LD,A	ADRESSE	Chargement du registre A avec le contenu mémoire définit par ADRESSE
INTFLO		Conversion $A_{Entier} \rightarrow A_{Flottant}$
ST,A		Chargement de la case mémoire définit par ADRESSE avec le contenu du registre A

Les 11 bits les plus significatifs de la microinstruction, forment le champ MCU, spécifié par la notice du constructeur:

- les 6 bits AC0-AC6 définissent le code de saut, nécessaire au calcul de l'adresse de la microinstruction suivante,
- les 4 bits FC0-FC3 contrôlent les indicateurs d'états qui permettent des débranchements conditionnés,
- le bit LD permet d'adresser directement une case de la mémoire de microprogramme à partir du bus mémoire.

La quasi-totalité des sauts sont relatifs à l'adresse de la microinstruction courante ou à une combinaison de cette adresse, soit avec le code opération, soit avec un des indicateurs d'états (cf annexe B).

5.3. Les instructions

5.3.1. Format

Le format des instructions peut être de un ou plusieurs mots de 24 bits. Le premier mot contient le code opération codé sur 8 bits (256 codes possibles), et une zone complémentaire adresse ou donnée de 16 bits. Les mots suivants sont des données, des adresses ou des constantes codées sur 24 bits.

Le programmeur dispose de 4 registres:

- deux accumulateurs A, B,
- deux registres de base pour l'indexation D, E.

5.3.2. L'adressage

L'adressage peut être de 4 types: direct, indirect, indexé et immédiat. Le bit de poids fort de la zone complémentaire (bit 15), associé au bit opérateur IT, permet un débranchement du microprogramme vers la séquence d'adressage indirect lorsque l'instruction autorise l'adressage direct et indirect, ou le choix du registre d'index lorsque l'adressage est indexé.

Grâce à la structure 24 bits, on dispose de 2 types d'adressage immédiat :

- court, sur 1 mot : l'adresse est codée sur les 16 bits de la zone complémentaire,
- long, sur 2 mots : l'adresse est codée sur les 24 bits d'un deuxième mot.

Le déroulement d'une instruction comporte 4 phases :

1. Recherche du code opération de l'instruction en mémoire centrale à l'adresse définie par le compteur ordinal et remise à jour du compteur.
A la fin de cette phase de recherche, le MCU examine si une requête d'interruption est présente : si oui, le déroulement normal de l'instruction est interrompue et un microprogramme de changement de contexte est exécuté avant le retour en phase de recherche de l'instruction à l'adresse définie par la nouvelle valeur du compteur ordinal.
2. Débranchement primaire du microprogramme vers une des 16 classes d'instructions, définie par les 4 bits de poids fort du code opération, par un saut JPX.
L'exécution de ce saut entraîne la mémorisation simultanée des 4 bits de poids faible du code opération dans un registre interne du MCU, de façon à libérer le bus mémoire tout en conservant l'information nécessaire aux débranchements secondaires.
3. Après une séquence commune aux instructions d'un même groupe, le décodage de 2 des 4 bits de poids faible du code opération permet un premier débranchement secondaire partiel vers 4 sous-groupes (JLL ou JRL). Ces 4 sous-groupes peuvent par un deuxième débranchement partiel mener aux séquences propres à chaque code opération selon la valeur de 2 bits restant à décoder (JRL ou JLL) dans le code opération.
Le débranchement secondaire peut également s'effectuer directement vers les séquences propres à chaque instruction par un décodage simultané des 4 bits de poids faible JPR (cf. fig.23).
4. En fin d'instruction toutes les branches se regroupent pour la phase de recherche de l'instruction suivante

La figure 24 donne le synoptique de déroulement d'une instruction.

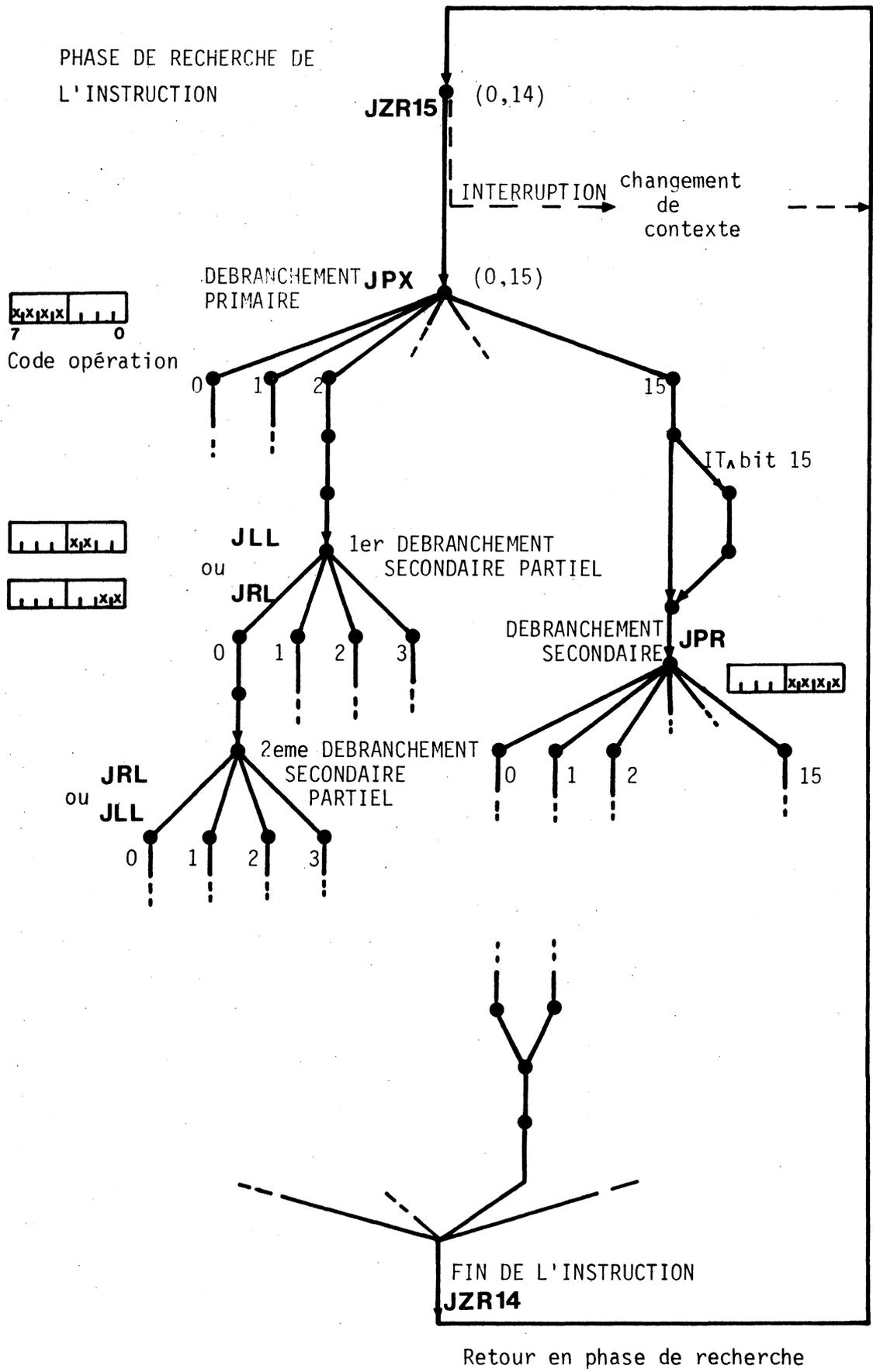


Figure 23 : Synoptique du décodage de l'instruction

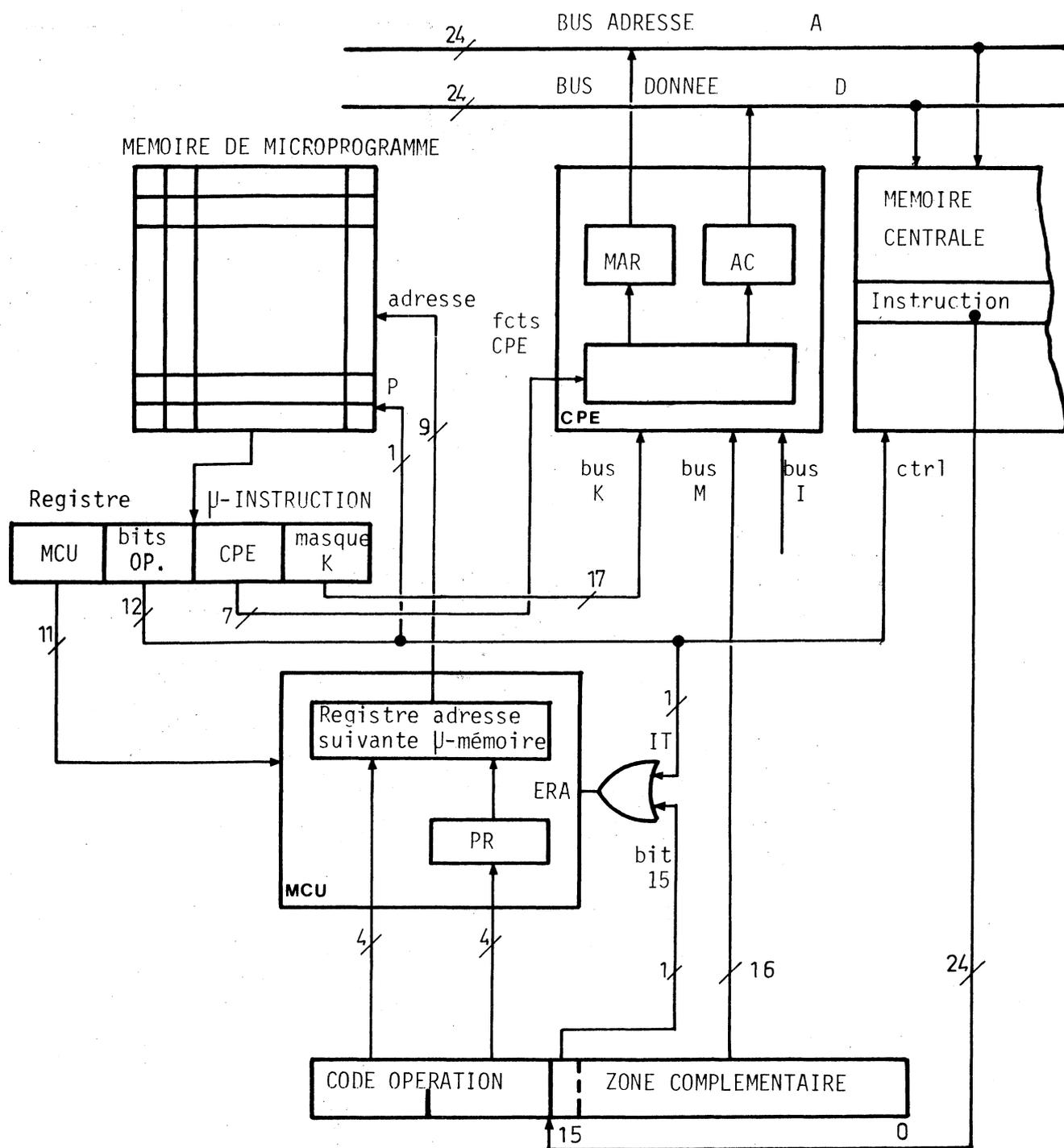


Figure 24 : Synoptique du déroulement d'une instruction

5.4. Le jeu d'instructions GESPRO

Le jeu d'instructions GESPRO se compose de deux groupes d'instructions distinctes :

- les instructions banalisées que l'on retrouve dans la plupart des jeux d'instructions classiques, enrichies d'instructions faisant utilisation de l'architecture spécialisée de GESPRO (interruptions et multiplieur)
- les instructions spécialisées, uniquement dédiées à l'expérience en cours, qui exécutent les algorithmes de filtrage du programme d'acquisition GESPRO.

1. Les instructions banalisées comportent :

- des instructions à référence mémoire : lecture-écriture mémoire avec possibilité d'adressage direct, indirect ou indexé, opérations logiques et arithmétiques entre accumulateurs et mémoire;
- des instructions à référence interne : opérations arithmétiques et logiques entre registres, transfert registre-registre, opérations de décalage et de rotation;
- des instructions de débranchements : sauts conditionnels et inconditionnels, et saut avec marquage autorisant l'utilisation de sous-programmes;
- des instructions multiplieur : multiplication entière et flottante, conversions entier-flottant et flottant-entier, addition flottante;
- des instructions de traitement des interruptions: activation et désactivation du module d'interruption et des interruptions, masquage-démasquage des niveaux d'interruptions, lecture du niveau courant;
- de microprogrammes "système" : initialisation de GESPRO et changement de contexte.

2. Les instructions spécialisées sont énumérées et explicitées dans le chapitre décrivant l'organigramme du programme de filtrage.

(Le jeu d'instructions est donné en Annexe C)

CHAPITRE II

I. METHODOLOGIE DE PROGRAMMATION DU SYSTEME DE FILTRAGE

1. Les traitements liés aux signaux de contrôle

Les tâches de contrôle sont liées à l'architecture du système et n'ont aucune influence sur le filtrage des événements, d'un point de vue physique. Pour réduire au minimum le temps mort qu'elles introduisent dans le programme de filtrage, elles sont systématiquement microprogrammées.

2. Les traitements de sélection

Dans un premier temps, les algorithmes de filtrage sont écrits en assembleur pour faciliter leur mise au point et leur contrôle. Lorsqu'un algorithme est parfaitement défini et testé, le traitement est microprogrammé afin d'optimiser sa vitesse d'exécution.

Chaque traitement doit être un module indépendant et chaînable de façon à pouvoir modifier facilement l'ordre des sélections dans le programme de filtrage, au vu des résultats expérimentaux. Pour ce faire, on essaie de microprogrammer le traitement en une instruction. Si cela pose des problèmes d'écriture ou d'interruptibilité (les interruptions ne sont prises en compte qu'en fin d'instruction), le traitement est décomposé en une suite d'instructions, conservant la modularité du traitement.

La complexité de la microprogrammation implique que pour un traitement donné, les temps nécessaires à l'écriture et aux tests, sont beaucoup plus longs (d'un facteur 10 à 20) en mode microprogrammé qu'en mode programmé.

De ce fait, le recours à la microprogrammation n'est envisagé que lorsque le gain en vitesse d'exécution est conséquent. La microprogrammation d'un traitement est particulièrement efficace dans les cas suivants :

a. L'algorithme est complexe ou contient des boucles :

En mode programmé, un tel algorithme demande l'exécution d'un grand nombre d'instructions banalisées. Or, chaque instruction contient une phase de recherche, en mémoire du code opération de l'instruction suivante. Cette séquence nécessaire du déroulement du programme, mais sans action sur l'exécution de l'algorithme, introduit un temps mort dans le déroulement du traitement, entre 300 et 700 nsec par instruction, selon que la recherche en mémoire s'exécute en recouvrement ou non. En mode microprogrammé, ce temps mort disparaît. Une seule phase de recherche est exécutée en fin d'instruction et le temps d'exécution est réduit en moyenne d'un facteur 10.

b. L'algorithme comporte de nombreuses références-mémoires :

La possibilité de masquage des temps d'accès-mémoire par recouvrement n'existe pas pour un traitement programmé. L'exécution des tâches, spécifiques au traitement programmé est strictement séquentielle. Dans le cas de deux cycles mémoires successifs programmés (2 instructions avec référence à la mémoire), l'adresse, et éventuellement la donnée du second cycle mémoire, ne peuvent être positionnées qu'une fois la première instruction exécutée, et donc le premier cycle achevé, car phase de recherche détruit le contenu des registres adresse et mémoire.

En mode microprogrammé, l'adresse et la donnée du second cycle mémoire peuvent être positionnées pendant l'exécution du premier cycle mémoire.

Ainsi un transfert mémoire-mémoire demande :

* 2.6 μ sec en mode programmé

* 1.3 μ sec en mode microprogrammé sans recouvrement

et 0.8 μ sec en mode microprogrammé avec recouvrement

Le masquage du temps d'accès mémoire n'est pas toujours possible, mais en moyenne 60 % des appels mémoire microprogrammés s'exécutent en recouvrement. Pour la plupart des traitements microprogrammés, le masquage des temps d'accès mémoire ne permet de réduire le temps d'exécution total que d'environ 10 % car la microprogrammation réduit les appels à la mémoire au minimum, par l'utilisation de tous les registres internes.

c. L'algorithme exige un traitement sur plusieurs variables :

En mode programmé, quatre registres seulement sont accessibles au programmeur. Dès que le nombre de variables à traiter augmente, les sauvegardes

en mémoire sont nécessaires et demandent 1.1 μ s pour l'écriture, 1.5 μ s pour la lecture.

En mode microprogrammé, et après sauvegarde du contexte, tous les registres sont disponibles, et 11 variables sont utilisables avant d'avoir recours à la sauvegarde en mémoire. Si cette sauvegarde est exécutée en recouvrement, elle n'introduit dans le microprogramme que 2 microinstructions supplémentaires, soit 300 ns :

- Pointeur de pile \rightarrow Registre Adresse Mémoire / Pointeur de Pile + 1 \rightarrow Pointeur de Pile
- Registre \rightarrow Accumulateur / Ecriture

L'utilisation de tous les registres internes du CPE (excepté le compteur ordinal R_4), des registres d'entrées du multiplieur et du bus I permet de microprogrammer le calcul de masse d'un dimuon décrit précédemment, sans aucune sauvegarde mémoire (cf. l'instruction MASS chapitre II.3).

d. L'algorithme comporte de nombreux débranchements ou un débranchement multiple :

En mode programmé, chaque débranchement nécessite au minimum une instruction, sans action sur le traitement à exécuter, soit en moyenne 1 μ s par débranchement. Dès que le nombre de débranchements augmente dans un traitement, le temps d'exécution effectif de ce traitement disparaît devant le temps imparti aux différents débranchements.

Dans le cas d'un débranchement multiple, un chaînage associé à un adressage indexé permet d'effectuer des débranchements parallèles, mais cette méthode est relativement lourde.

En mode microprogrammé, les débranchements nécessitent au minimum une instruction, sans action sur le traitement à exécuter, soit en moyenne une 1 μ s par débranchement. Dès que le nombre de débranchements augmente dans un traitement, le temps d'exécution effectif de ce traitement disparaît devant le temps imparti aux différents débranchements.

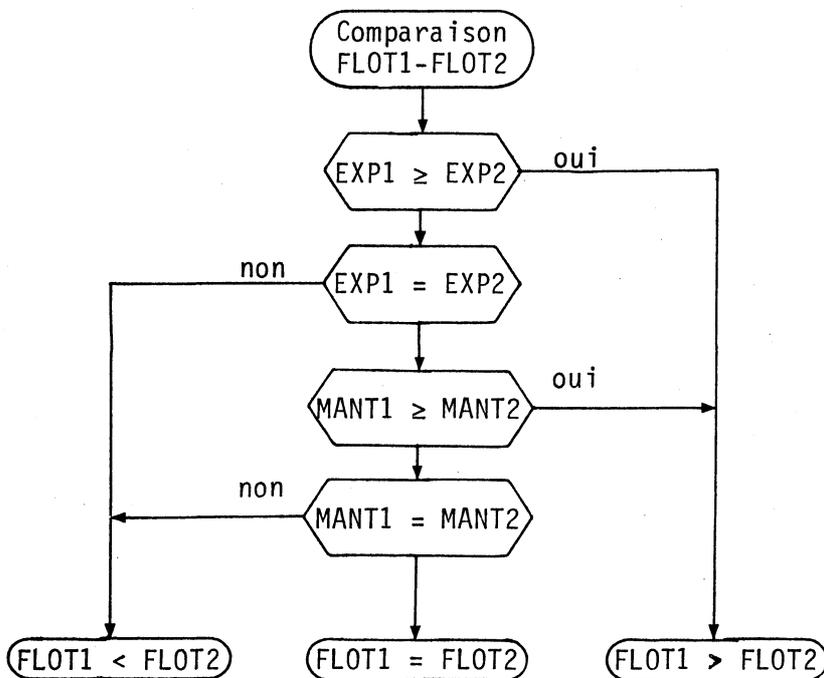
Dans le cas d'un débranchement multiple, un chaînage associé à un adressage indexé, permet d'effectuer des débranchements parallèles, mais cette méthode est relativement lourde.

En mode microprogrammé, les débranchements séquentiels peuvent se suivre toutes les microinstructions (150 nsec), et ne nécessitent pas toujours selon le test effectué, de microinstruction supplémentaire. Les débranchements

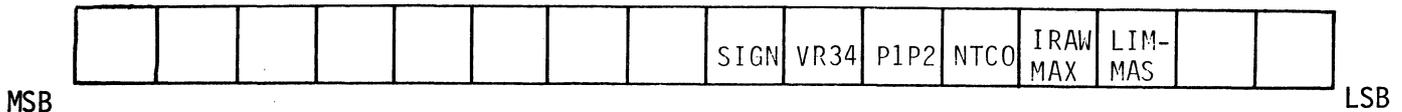
multiples parralèles s'exécutent aisement à l'aide d'un saut JPX, directement, si les débranchements sont conditionnés à une donnée mémoire, ou en utilisant un chaînage mémoire.

Pour exemple, la comparaison de deux nombres flottants positifs selon l'organigramme ci-dessous, demande, selon la branche de sortie:

- de 10 à 25µs en mode programmé,
- de 2.1 à 3.4µs en mode microprogrammé.(instruction CPMASS)



L'introduction d'un nouveau test de sélection peut entraîner une inefficacité du filtre par le rejet de bons candidats. Le filtre programmé, contrairement au filtre câblé, peut mesurer cette inefficacité en ne rejetant pas l'événement mais en le marquant, pour signaler que ce dernier ne satisfait pas au test. L'événement étant acquis, l'analyse hors ligne permet de vérifier si la décision GESPRO était justifiée. Si l'étude est satisfaisante, le test est alors programmé en mode rejet par activation du bit correspondant dans le mot de rejet (REJECT-WORD).



En cours d'acquisition, un échantillonnage d'événements non biaisés, (c'est à dire ne subissant aucun rejet de façon à être acquis) permet un contrôle continu de l'efficacité du filtrage. Cette option est très intéressante, car le contrôle est constant dans le temps au cours de l'acquisition. Lorsque cette option n'existe pas, il faut intercaler entre des séries d'acquisitions avec filtrage, des cycles d'acquisitions de données-tests sans filtrage, mais dans ce cas, test et filtrage ne sont pas aussi liés dans le temps. Les conditions d'expérience peuvent changer entre les acquisitions de données tests et les acquisitions de données filtrées ce qui réduit l'efficacité du contrôle.

Les événements non biaisés subissent les mêmes tests que les événements biaisés. Si un test est négatif, cette information est indiquée dans le statut de traitement de l'événement, et ce "rejet" n'est pas comptabilisé dans les histogrammes de rejet ou de marquage GESPRO, mais compté, quel que soit la branche de sortie du traitement, comme événement non biaisé.

Lors de l'analyse des données hors-ligne, le contrôle continu de la validité des décisions GESPRO peut être effectué au vu du statut de traitement transmis avec l'information totale de l'événement non biaisé.

1.2. Les tests de contrôle

De nombreux paramètres permettent de vérifier la validité de l'information à traiter. Le filtre programmable permet de conditionner le rejet ou la poursuite du traitement au type d'erreur détectée, contrairement au filtre câblé où une telle option est très difficile à réaliser.

Dans le programme d'acquisition GESPRO, toute détection d'erreur entraîne le marquage et le transfert de cet événement de façon à ne pas perdre l'information nécessaire à la localisation de la source de certaines erreurs (compteurs déficients, absence de marqueurs....)

1.3. L'histogrammatation

Pour chaque événement, GESPRO doit histogrammer :

- les signaux de contrôle (SDI, STA, GOOD ou BAD),
- toutes les branches de sortie : - en mode marquage,
- en mode rejet,

et le cas échéant, la combinaison de sextants qui a donné lieu à cet événement

- toutes les sorties en erreurs.

Cette information est contenue dans la mémoire centrale de GESPRO. Entre deux cycles d'acquisition, ces histogrammes sont lus en vol de cycle par le miniordinateur et visualisables par l'opérateur. En fin d'acquisition, les histogrammes définitifs sont sauvegardés sur bande à la suite des données acquises.

2. ORGANIGRAMME DU PROGRAMME DE FILTRAGE

La figure 25 représente l'organigramme du programme de filtrage utilisé en acquisition sur l'expérience NA10 depuis janvier 1982.

Les traitements marqués d'un signe μP sont des séquences entièrement microprogrammées qui seront décrites dans le chapitre suivant.

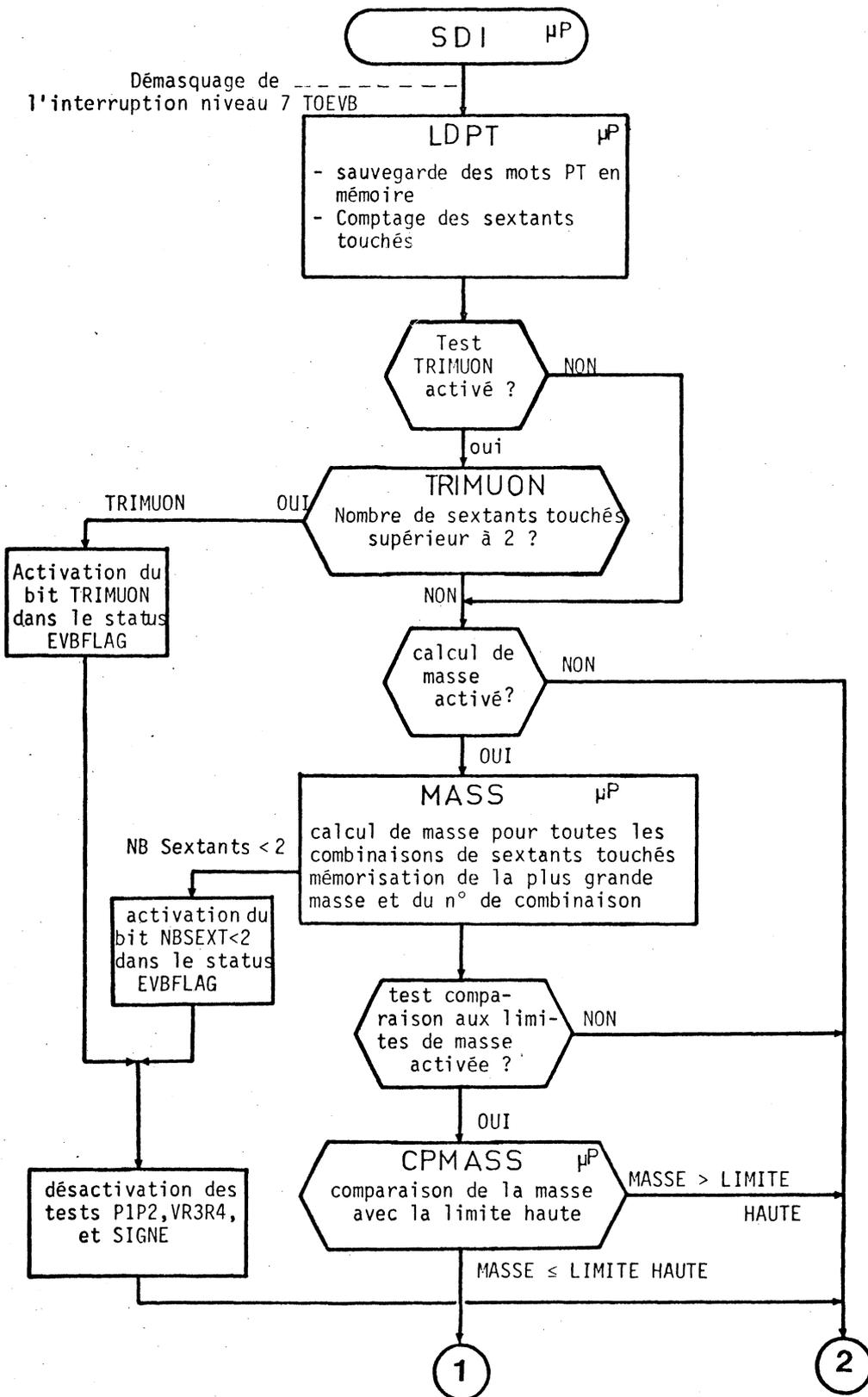
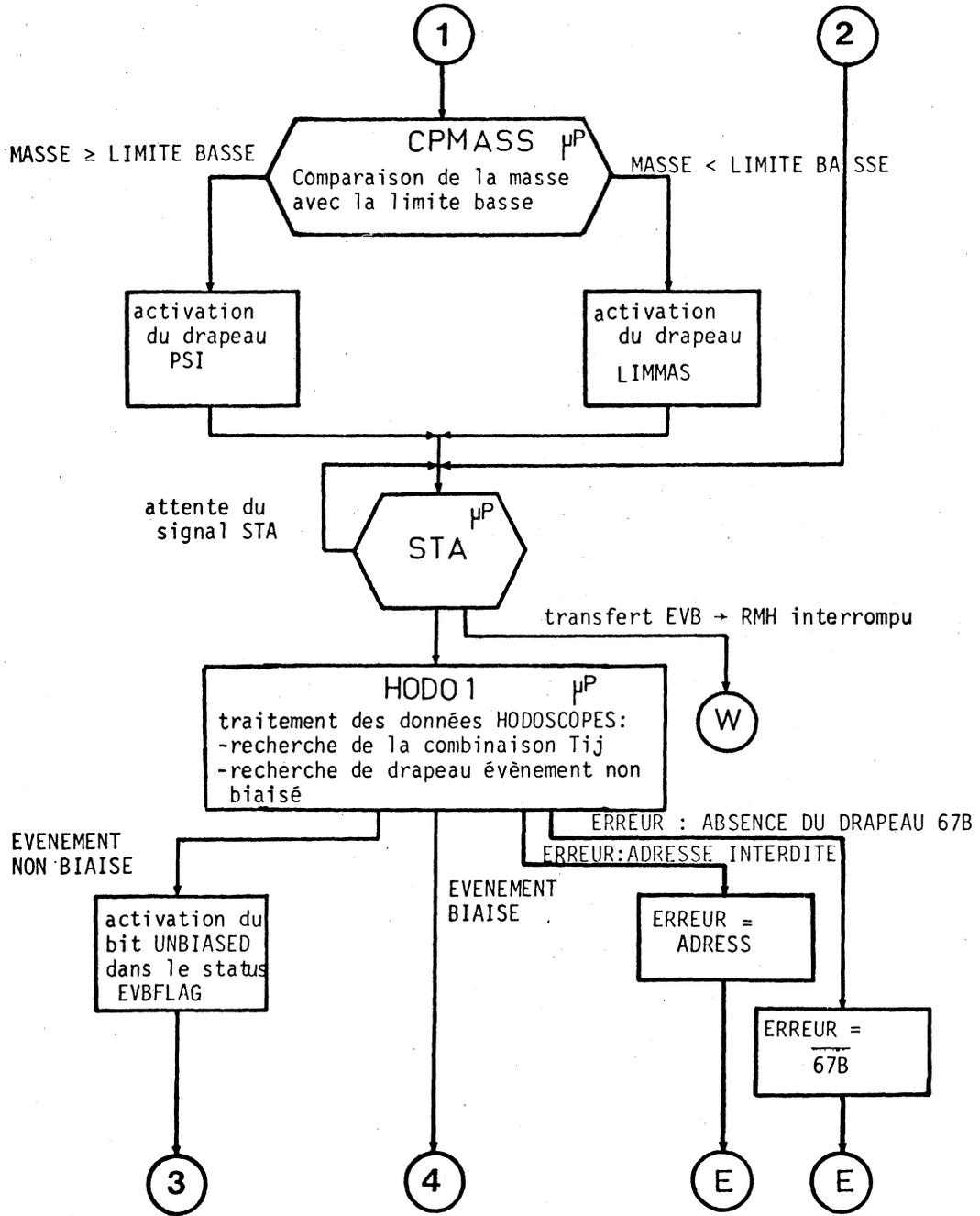
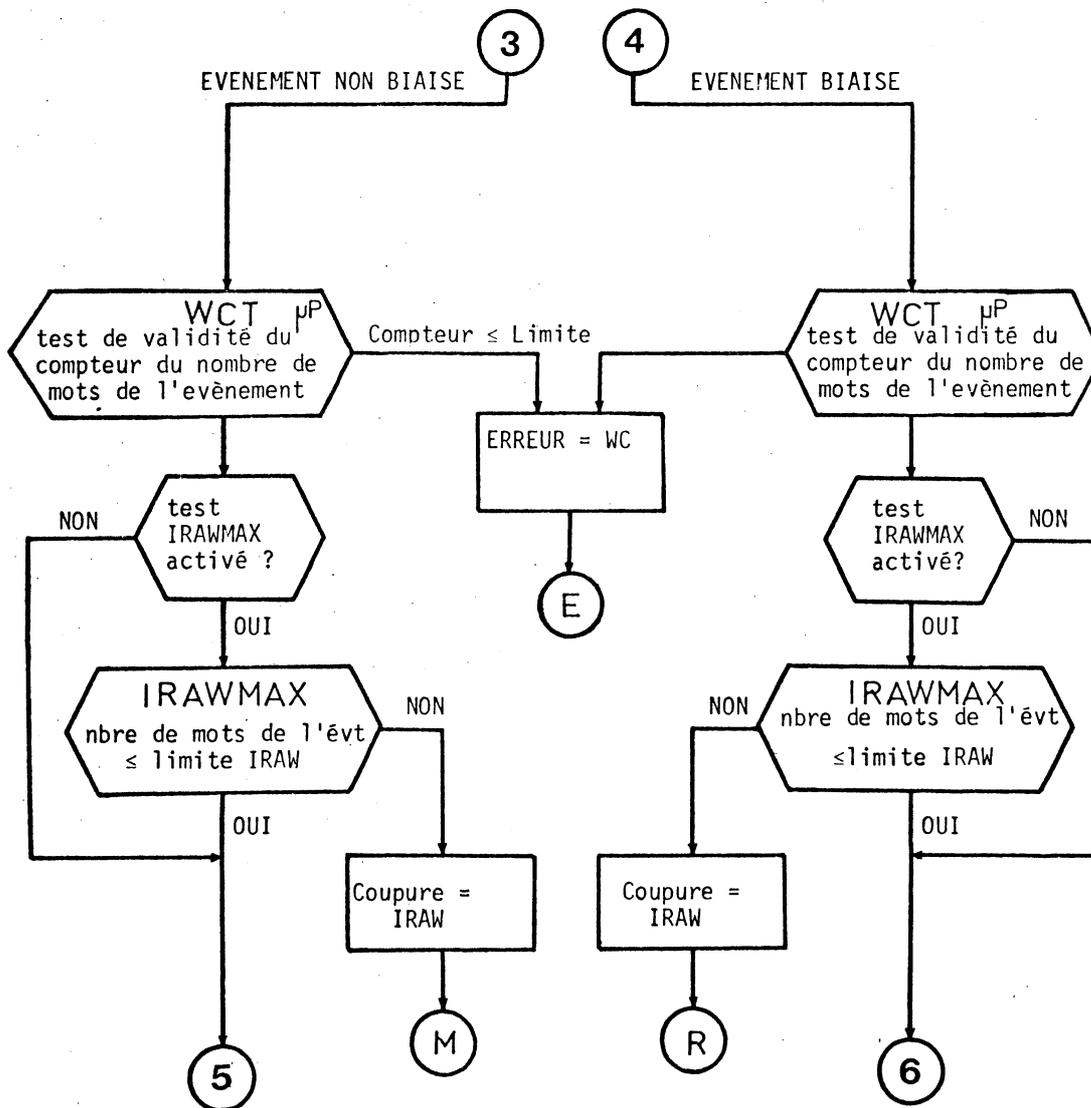
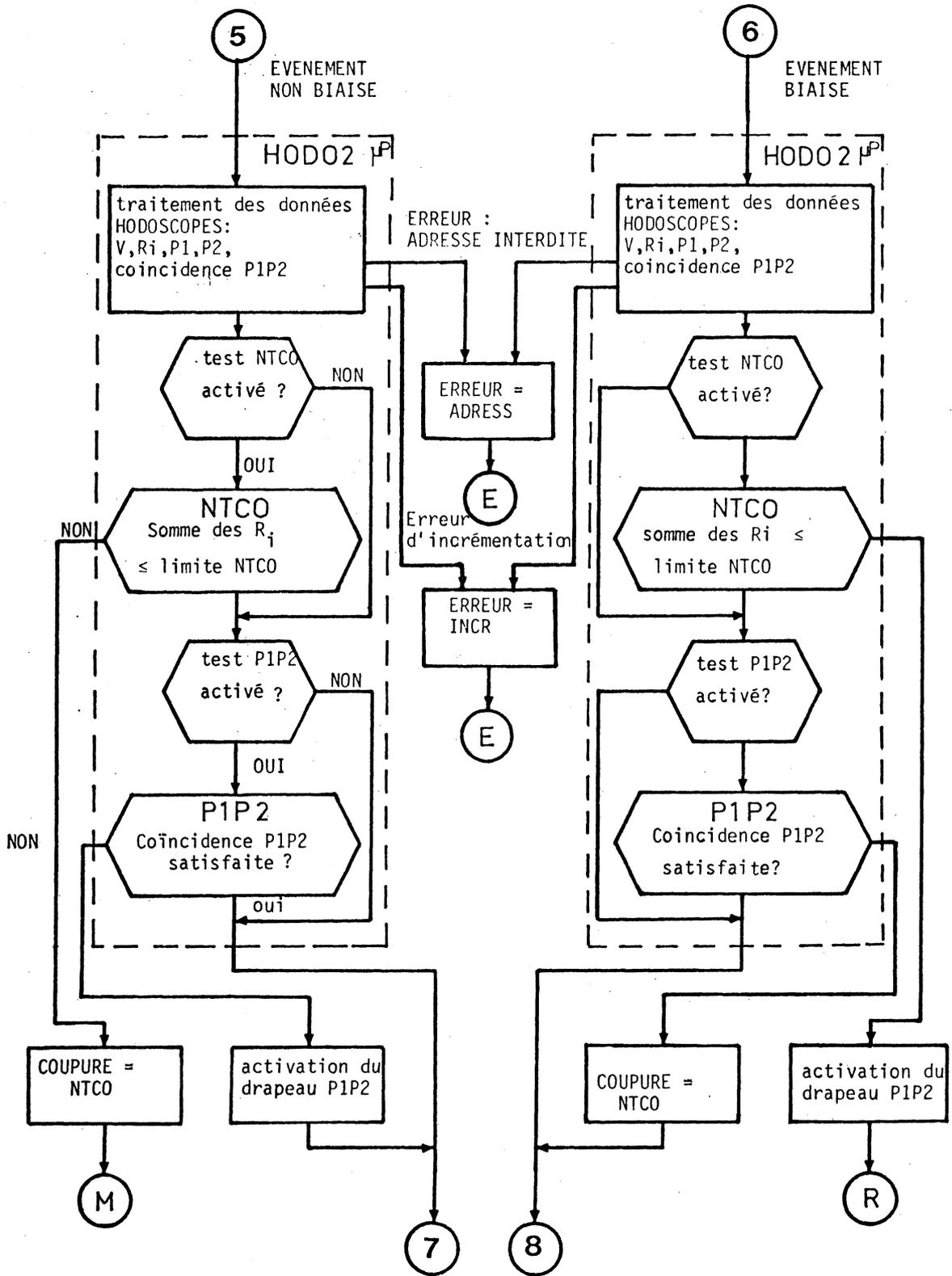
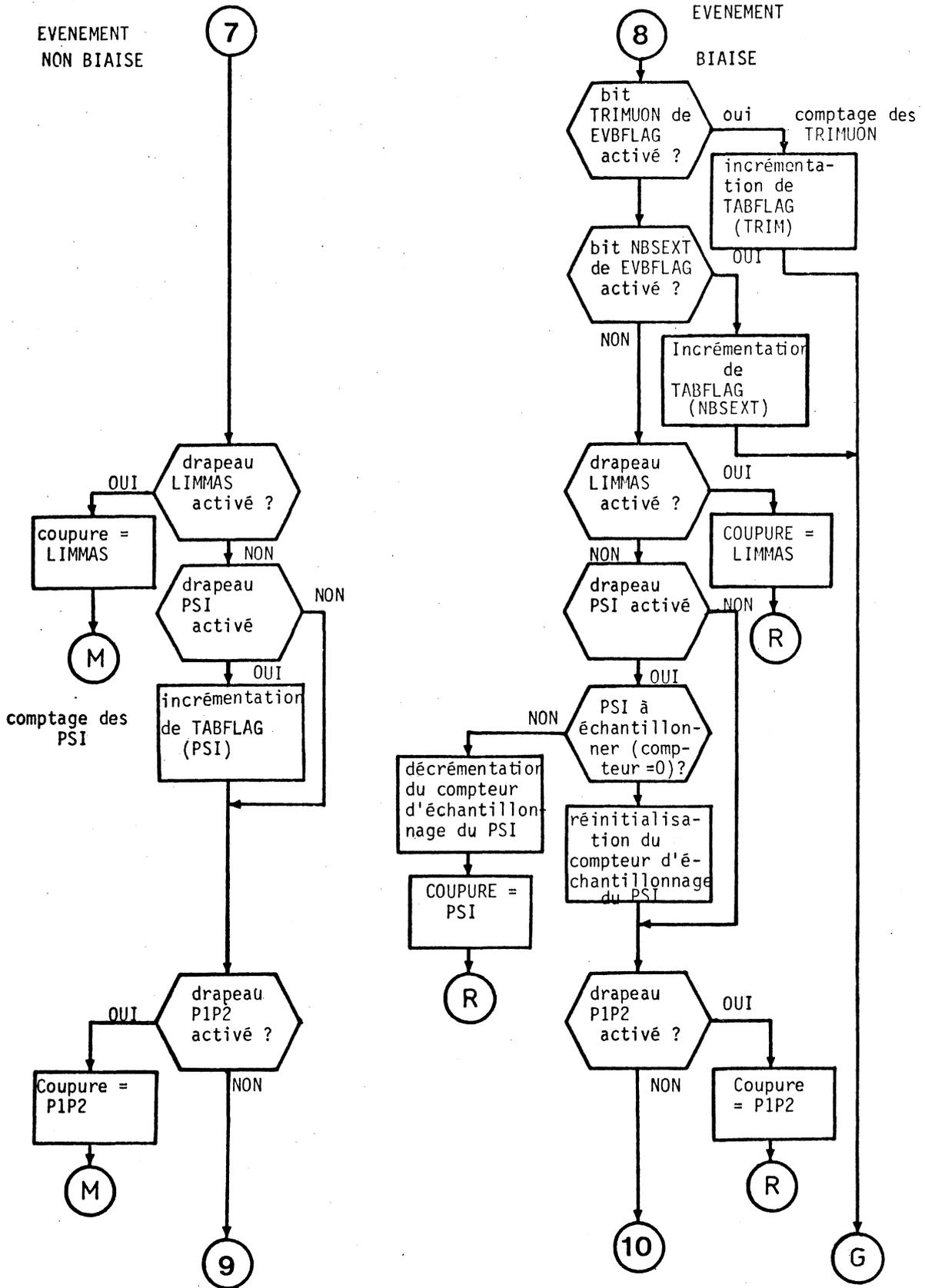


Figure 25: Organigramme du programme de filtrage NA10

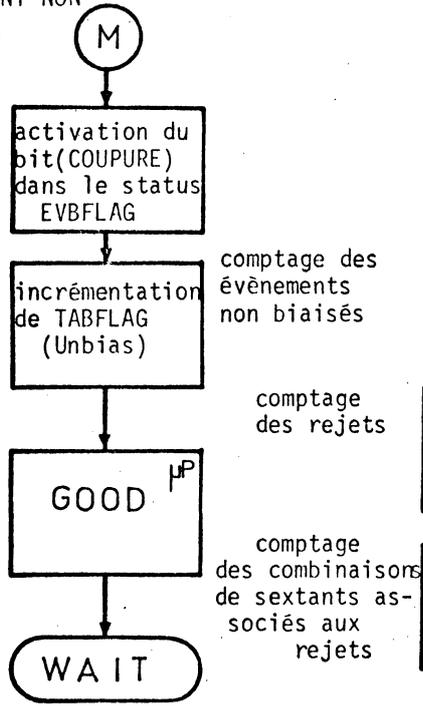




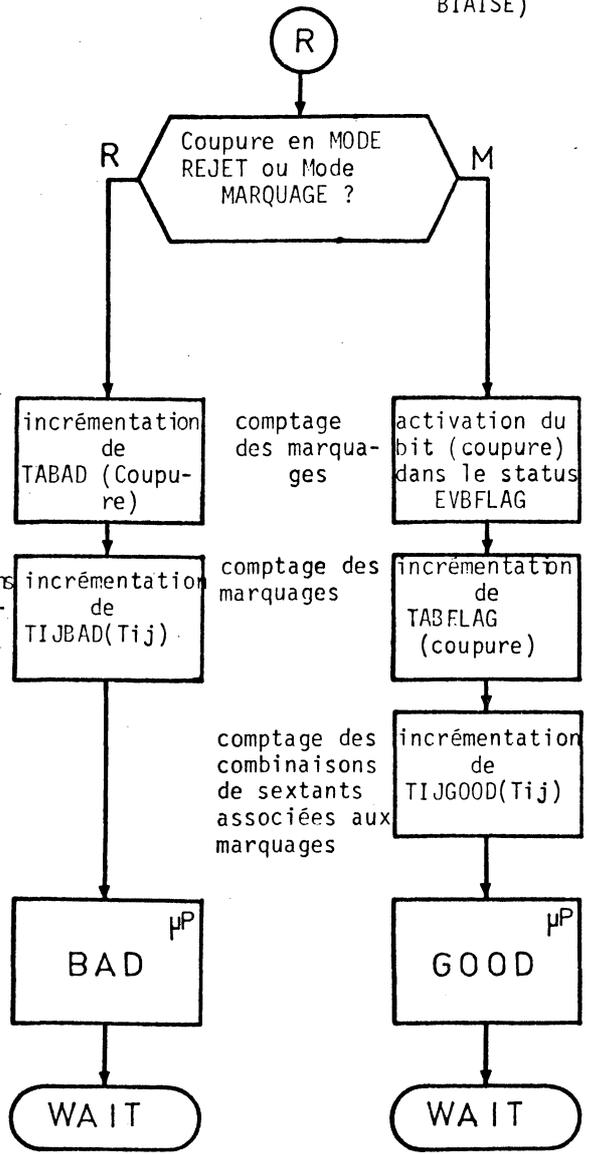




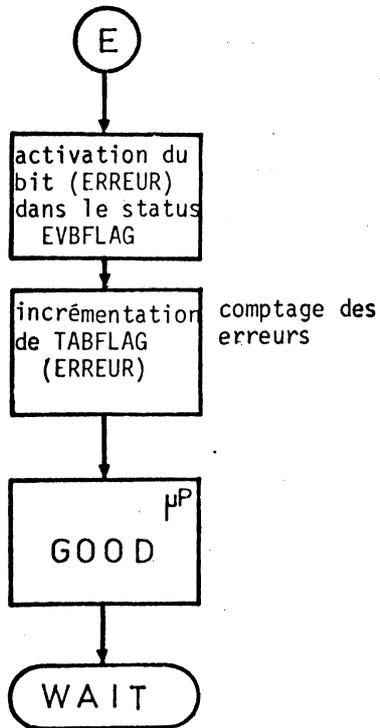
TRAITEMENT DES COUPURES
(EVENEMENT NON BIAISE)



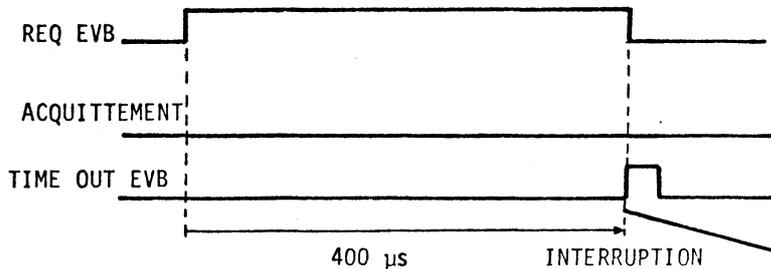
TRAITEMENT DES COUPURES (EVENEMENT BIAISE)



TRAITEMENT DES ERREURS

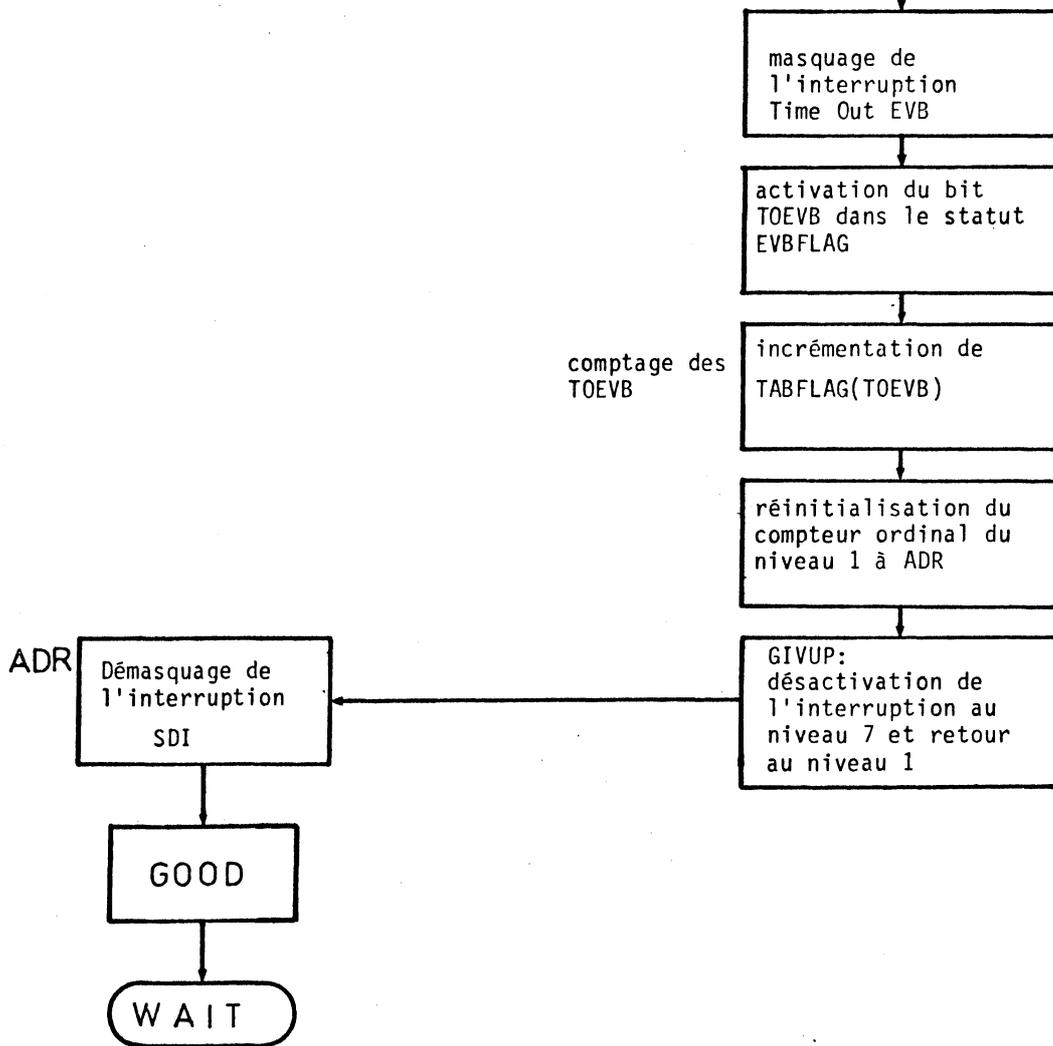


Niveau 1



INTERRUPTION
masquage SDI et
changement de contexte
Niveau 1 -> Niveau 7

Niveau 7



2.1. Le traitement sur le résumé de données

Pendant le transfert des données de l'événement dans l'EVB, GESPRO exécute les traitements liés aux moments transverses, délivrés par les cartes de P_T .

SDI est la séquence de lancement du programme de filtrage GESPRO. A la fin de l'exécution de cette séquence, la recherche des plus grands moments transverse est achevée et GESPRO transfère les mots P_T en mémoire par l'instruction LDPT, avant que les opérateurs ne soient alloués à une autre unité GESPRO. Simultanément, l'instruction LDPT compte le nombre de sextants touchés

Le test TRIMUON permet de marquer les événements ayant plus de deux sextants touchés et le calcul de masse n'est pas exécuté car la coupure en masse n'est pas effectuée sur ces événements.

L'instruction MASS calcule à partir des valeurs des plus grands moments transverses de chaque sextant, la masse du dimuon pour chaque combinaison de sextants touchés. Elle ne conserve que la plus grande masse calculée et la combinaison S_{ij} qui y a donné lieu.

Une valeur limite de P_T peut être appliquée de façon à ne considérer comme touchés, que les sextants ayant un moment transverse supérieur ou égal à cette limite. Si tous les sextants sont touchés 15 calculs doivent être effectués. Si l'événement n'a qu'un sextant touché, le calcul de masse ne peut pas être effectué et le programme se poursuit par la branche NBSEXTANTS < 2. Cette impossibilité de traitement est signalée par un bit du mot statut de l'événement (EVB FLAG).

Lorsque le calcul de masse n'est pas exécuté, le programme désactive les 3 tests qui sont fonctions de la combinaison de sextants fournie par le calcul de masse : P_1P_2 , VR_3R_4 et SIGNE.

La comparaison CPMASS entre la masse et les limites programmées en début d'acquisition permet de classer l'événement dans 3 catégories :

- masse inférieure à la limite basse : événement de faible masse
- masse comprise entre les 2 limites : événement de masse intermédiaire (PSI)
- masse supérieure à la limite haute : événement de haute masse.

Les événements de faible masse sont signalés par un bit du statut, de même que les événements de masse intermédiaire.

L'absence de l'un de ces deux bits signifie que l'événement est de haute masse.

Dans cette version de traitement, la coupure en masse n'a pas été retenue comme premier critère de rejet car le système d'acquisition doit conserver les événements de masse intermédiaire à des fins d'analyse. Les événements de faible masse ne représentent quant à eux, que quelques pourcents, comme le montreront les résultats exposés dans un chapitre suivant.

2.2. Le traitement sur les données acquises :

Après STA, GESPRO dispose de toutes les données de l'événement, mais cette version du programme de filtrage ne traite que les données hodoscopes.

Une première instruction HODO 1 mémorise la combinaison de sextants délivrée par la logique intersextants. Cette instruction détermine, si l'événement est biaisé ou non biaisé, par la reconnaissance d'un marqueur fournie par la logique intersextants. Cette détermination doit être faite avant les tests de filtrage.

L'absence du flag de début de branche hodoscope (67B) ou une adresse compteur interdite sont considérées comme des erreurs et interrompent le traitement : l'erreur est signalée dans le status, et l'événement est transmis au mini-ordinateur. Qu'il s'agisse d'événements biaisés ou non, les erreurs sont traitées identiquement.

Avant d'effectuer un test sur la taille de l'événement, la validité de l'information contenue dans le compteur du nombre de mots de l'événement est testée par l'instruction WCT. Devant chaque événement, un en-tête de 36 mots est écrit systématiquement. La valeur du compteur doit donc être supérieure à cette valeur limite pour être validée.

Le test IRAWMAX exécute une coupure sur la taille des événements, car l'analyse a montré que les événements de taille supérieure à 800 mots sont des événements de haute multiplicité produits par le bruit de fond et irréconstructibles dans la quasi-totalité des cas pour les programmes d'analyse.

L'instruction HODO 2 traite la suite des données hodoscopes. Les traitements effectués dans les deux instructions HODO 1 et HODO 2 ont été séparés pour permettre l'exécution du test IRAWMAX sans attendre la fin du traitement du bloc de données hodoscopes. On gagne ainsi en temps de traitement, car IRAWMAX a un meilleur facteur de réjection que le test NTCO effectué dans HODO 2 sur les données hodoscopes.

Le traitement HODO 2 :

exécute une coupure (NTCO) sur la multiplicité des compteurs R_i

- prépare les tests suivants en mémorisant dans un tableau les différentes valeurs des compteurs R_3 et R_4 touchés pour chaque sextant, de même que les compteurs P_1 et P_2 .

Des adresses compteurs non incrémentales ou interdites sont des erreurs, traitées à travers les branches de sortie correspondantes.

A la fin du traitement des données hodoscopes, l'instruction HODO 2 exécute le test P_1P_2 , qui vérifie que la coïncidence des compteurs P_1 et P_2 est réalisée, pour la combinaison de sextants touchés par le dimuon de plus haute masse.

La mise en coïncidences des compteurs $P_1 * P_2$ permet d'éliminer les coïncidences accidentelles de traces qui ne sont pas contenues dans le même plan φ (plan de déflexion de la particule dans l'aimant).

En cours d'élaboration du programme de traitement, d'autres tests sont venus s'intercaler entre les tests NTCO et P_1P_2 alors que le microprogramme de l'instruction HODO 2 était déjà implantée. Le résultat du test de coïncidence P_1P_2 est mémorisé et sera examiné plus en avant dans le programme de filtrage.

Si l'événement est non biaisé et qu'il s'agit d'un trimuon ou que le nombre de sextants touchés est inférieur à 2, la suite du traitement liée à l'exécution de l'instruction MASS ne concerne pas ces événements et le traitement est terminé.

Si l'événement est biaisé, et qu'il s'agit d'un trimuon ou que le nombre de sextants touchés est inférieur à 2, le bit correspondant du mot statut a été activé dans la branche TRIMUON ou NBSEXTANT, et le programme passe directement au test sur la valeur de la masse.

Plusieurs rejets sont possibles sur les valeurs de la masse :

- a) tous les événements sont conservés mais les événements de faible masse ($M < L$) où les événements de masse intermédiaire ($L \leq M \leq H$) sont marqués dans le statut,
- b) les événements de faible masse sont tous rejetés. Un facteur d'échantillonnage permet de conserver une fraction des événements de masse intermédiaire. Ce facteur d'échantillonnage est programmable en début d'acquisition.

Le résultat du test de coïncidence P_1P_2 est examiné avant d'exécuter l'instruction VR_3R_4 qui teste la condition d'alignement des compteurs V , R_3 , R_4 (mentionnée dans la description du principe de déclenchement). Pour des raisons technologiques, certaines coïncidences $V * R_3 * R_4$ interdites sont autorisées par la matrice de coïncidence et le traitement VR_3R_4 permet de les éliminer en vérifiant que pour les deux sextants touchés par le dimuon de plus grande masse, existe un triplet (VR_3R_4) satisfaisant aux conditions d'alignement. Dans le cas contraire,

l'événement est rejeté.

L'instruction VR_3R_4 contient un dernier test SIGNE qui permet de ne retenir que les dimuons formés de deux muons de signes contraires mais cette coupure n'a jamais été activée jusqu'à présent en cours d'expérience.

2.3. Le traitement du "dépassement en temps" EVB

Si un appel GESPRO à la mémoire EVB n'est pas acquitté, le calculateur émet un signal de non-acquittement TOEVB. Ce signal est appliqué à l'entrée de l'interruption externe GESPRO de niveau 7. Le programme normal s'exécutant au niveau 1, l'interruption est prise en compte, dès le retour en phase de recherche. Le changement de contexte masque les interruptions SDI, de façon à ne pas pouvoir être interrompu pendant l'exécution du programme d'erreur TOEVB au niveau 7. A la fin de ce traitement, GESPRO désactive l'interruption et redescend au niveau 1, pour démasquer l'interruption SDI et émettre une réponse GOOD comme toutes les autres branches de sortie en erreur.

3. Les instructions spécialisées MASS et HODO 2

Ce chapitre décrit les deux instructions les plus spécifiques du programme de filtrage.

3.1. L'instruction MASS : Calcul de la masse des dimuons

L'instruction MASS utilise tous les opérateurs câblés développés autour du noyau GESPRO : les cartes de recherche des plus grands moments transverse et toutes les fonctions de la carte multiplieur.

A partir des valeurs de moment transverse délivrées par les cartes de recherche du plus grand PT, l'instruction calcule la masse du dimuon pour toutes les combinaisons de sextants touchés et, délivre la plus grande masse calculée et la combinaison de sextants correspondante. La figure 26 visualise les différentes tables et leur chaînage. Les figures 27 et 28 donnent l'organigramme du microprogramme.

1. Génération de TABVAL

A partir des mots P_T , le microprogramme génère dans la table TABVAL, pour chaque sextant touché, les variables nécessaires au calcul de masse :

- $\sin\theta$ et $\cos\theta$ sont fournis par la table TABV (64 mots) en fonction de la valeur associée au mot P_T ,
- l'impulsion transverse P_i est recherchée dans la table TABPI (1024 mots) en fonction du couple de valeurs (V, P_T) .

Simultanément à la génération de TABVAL, le microprogramme génère un mot status des sextants touchés SEXTOU, codé sur 6 bits, à raison d'un bit par sextant.

2. Recherche des paramètres :

A partir du code de sextants SEXTOU, le microprogramme recherche une table intermédiaire, l'adresse qui permet de pointer dans la table des combinaisons TABMENU, sur le début de la liste des combinaisons S_{ij} pour lesquelles la masse du dimuon doit être calculée. Un indicateur permet au microprogramme de savoir s'il pointe ou non sur la fin de la liste.

A chaque combinaison S_{ij} , sont associées 4 données fournies par la table

TABPT : un code TRIGAD de la combinaison de sextants, les 2 adresses des sextants correspondants dans la table TABVAL et le terme $\cos(\varphi_1 - \varphi_2)$, constante liée à la combinaison de sextants considérés.

3. Le calcul de masse pour une combinaison de sextants :

Quand les données nécessaires sont définies, le calcul de masse du dimuon issu de ces 2 sextants peut s'exécuter selon la formule :

$$-\frac{M^2}{2} = P_{i_1} P_{i_2} (\cos\theta_1 \cos\theta_2 + \sin\theta_1 \sin\theta_2 \cos(\varphi_1 - \varphi_2) - 1)$$

4. Comparaison de la masse

Cette valeur de la masse est comparée à la valeur de la plus grande des masses calculées précédemment pour d'autres combinaisons. Si la nouvelle masse calculée est supérieure, le programme la sauvegarde, ainsi que le code TRIGAD de la combinaison de sextants associée.

Le microprogramme réexécute la même séquence pour une nouvelle combinaison de sextants définie par la liste de combinaisons à exécuter, ou retourne en phase de recherche, si tous les calculs de masse ont été exécutés.

La masse du dimuon est calculée par GESPRO en 13.0 μ s. Par comparaison, le miniordinateur ND10 demande pour le même calcul 120 μ s.

Le temps d'exécution total de l'instruction est de :

- 37.6 μ sec pour 2 sextants touchés
- 78.3 μ sec pour 3 sextants touchés
- 138.3 μ sec pour 4 sextants touchés
- 218.1 μ sec pour 5 sextants touchés
- 317.0 μ sec pour 6 sextants touchés

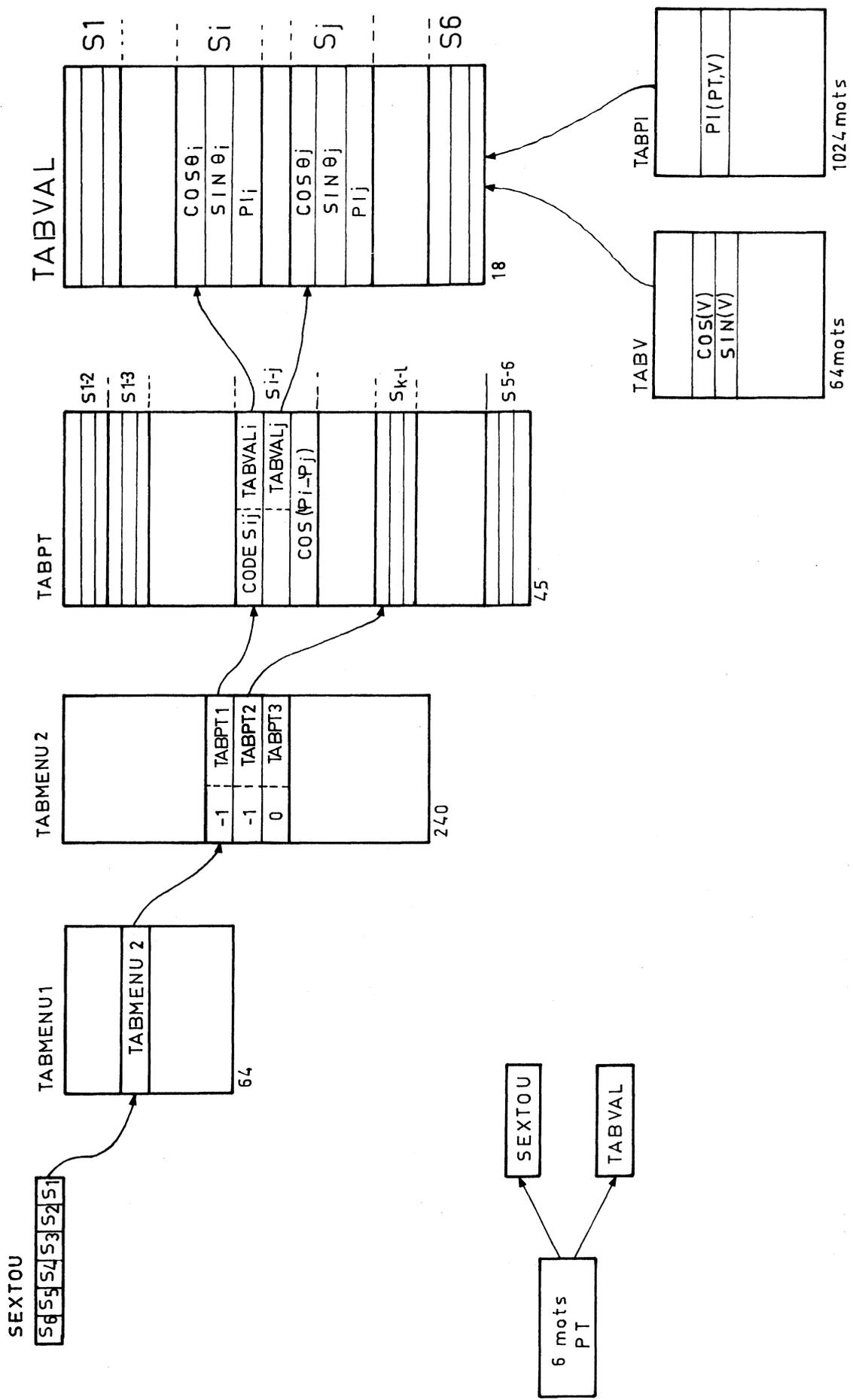


Figure 26 : Table de chaînage de l'instruction MASS

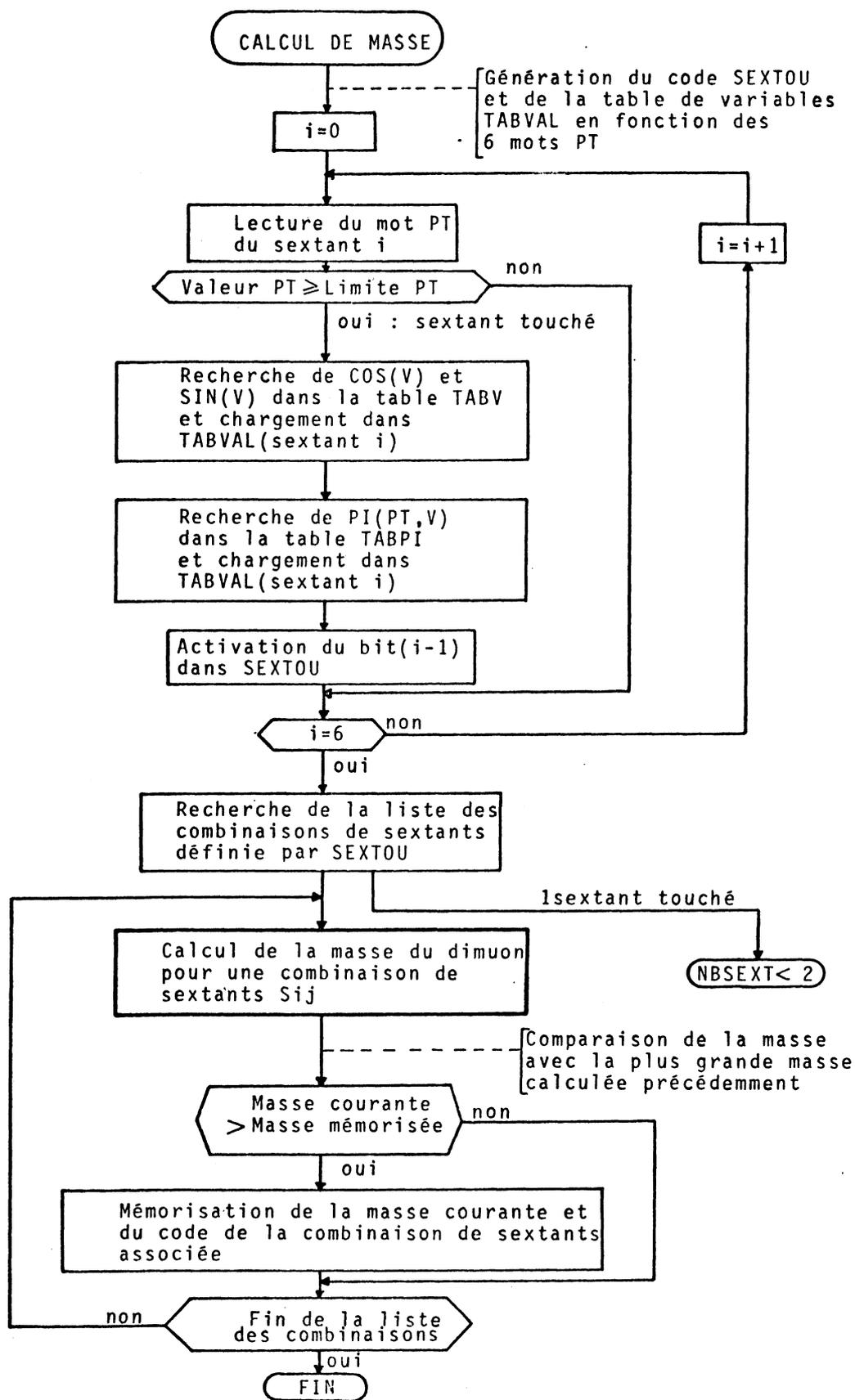


Figure 27: Organigramme de l'instruction MASS

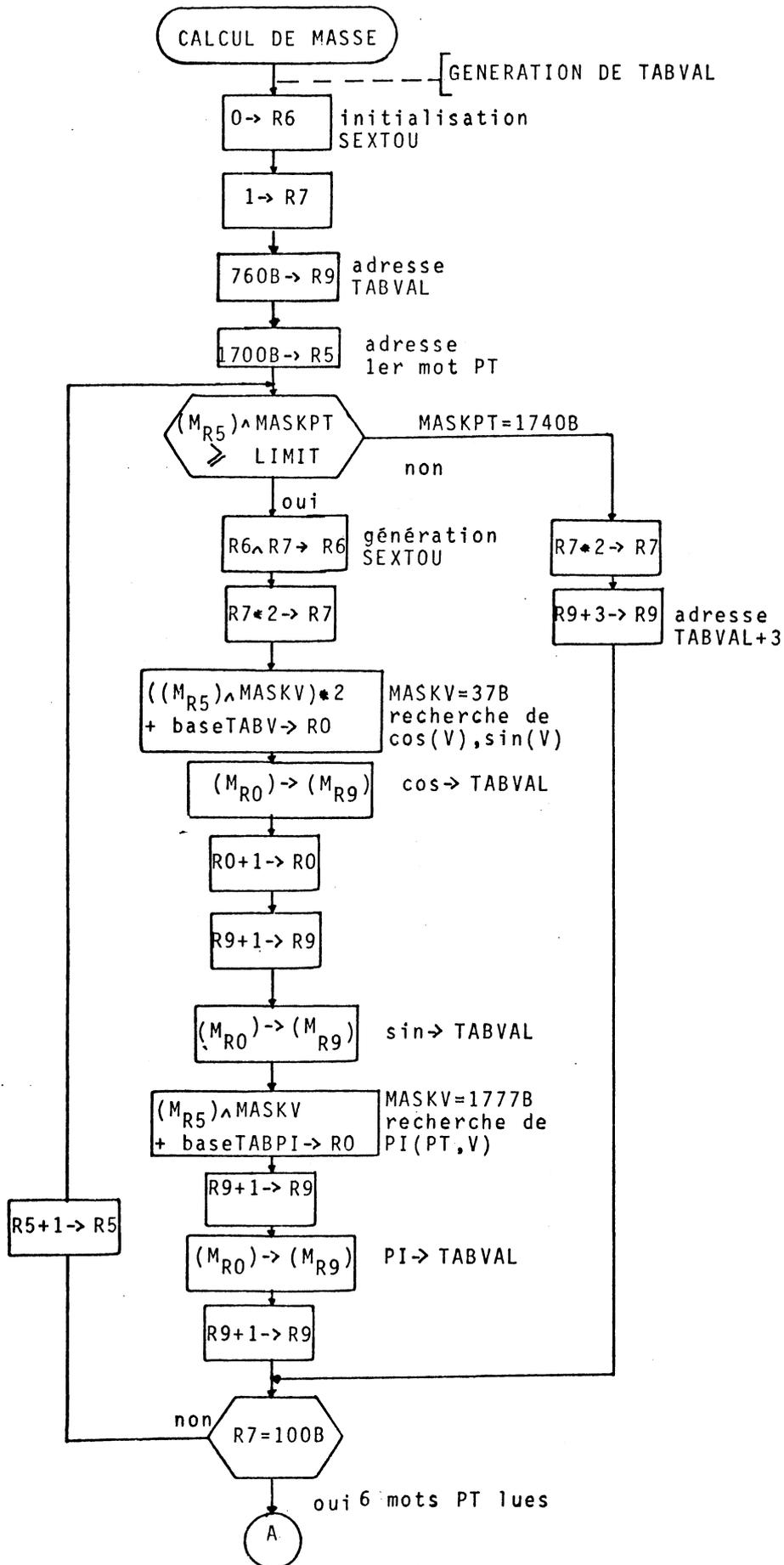
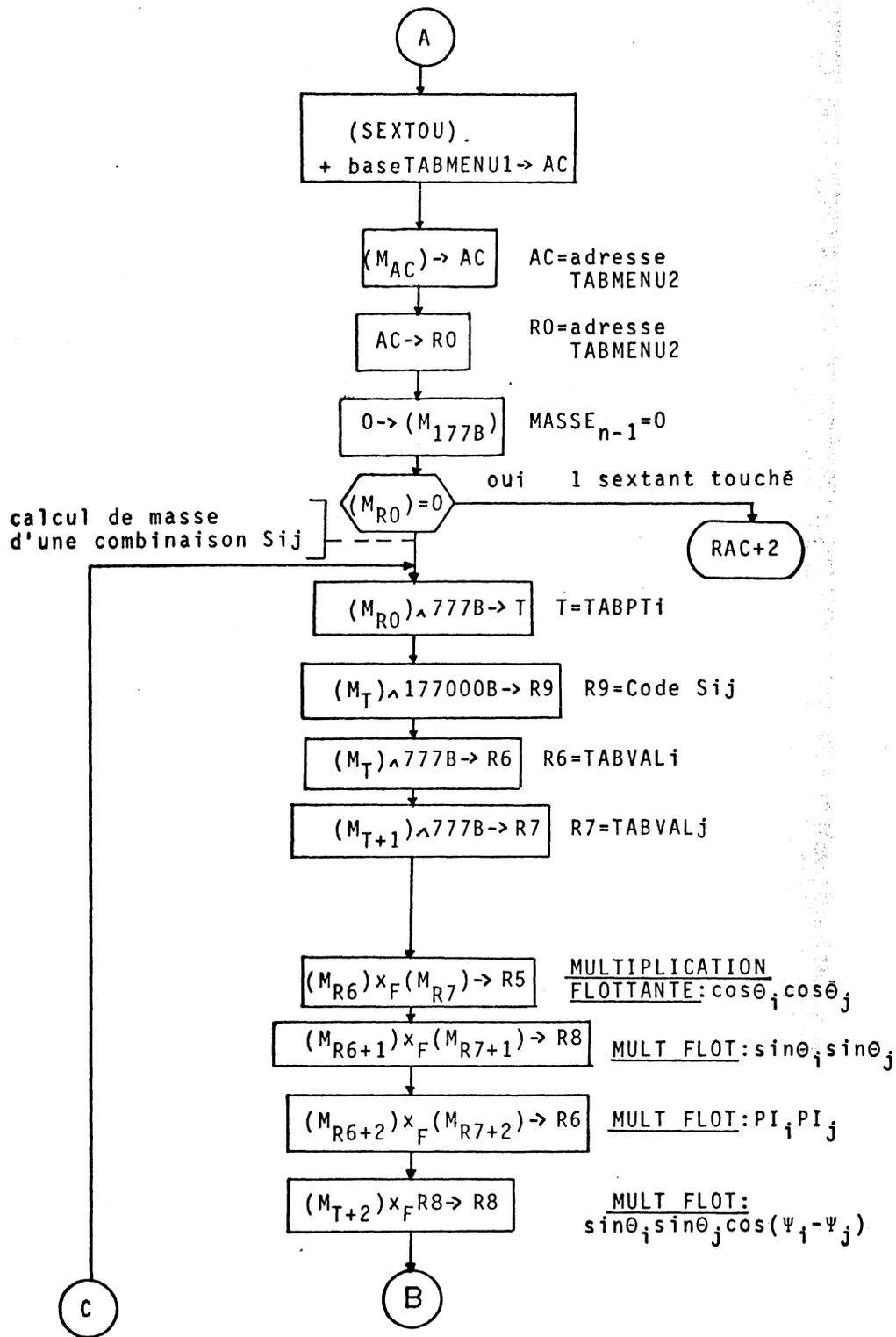
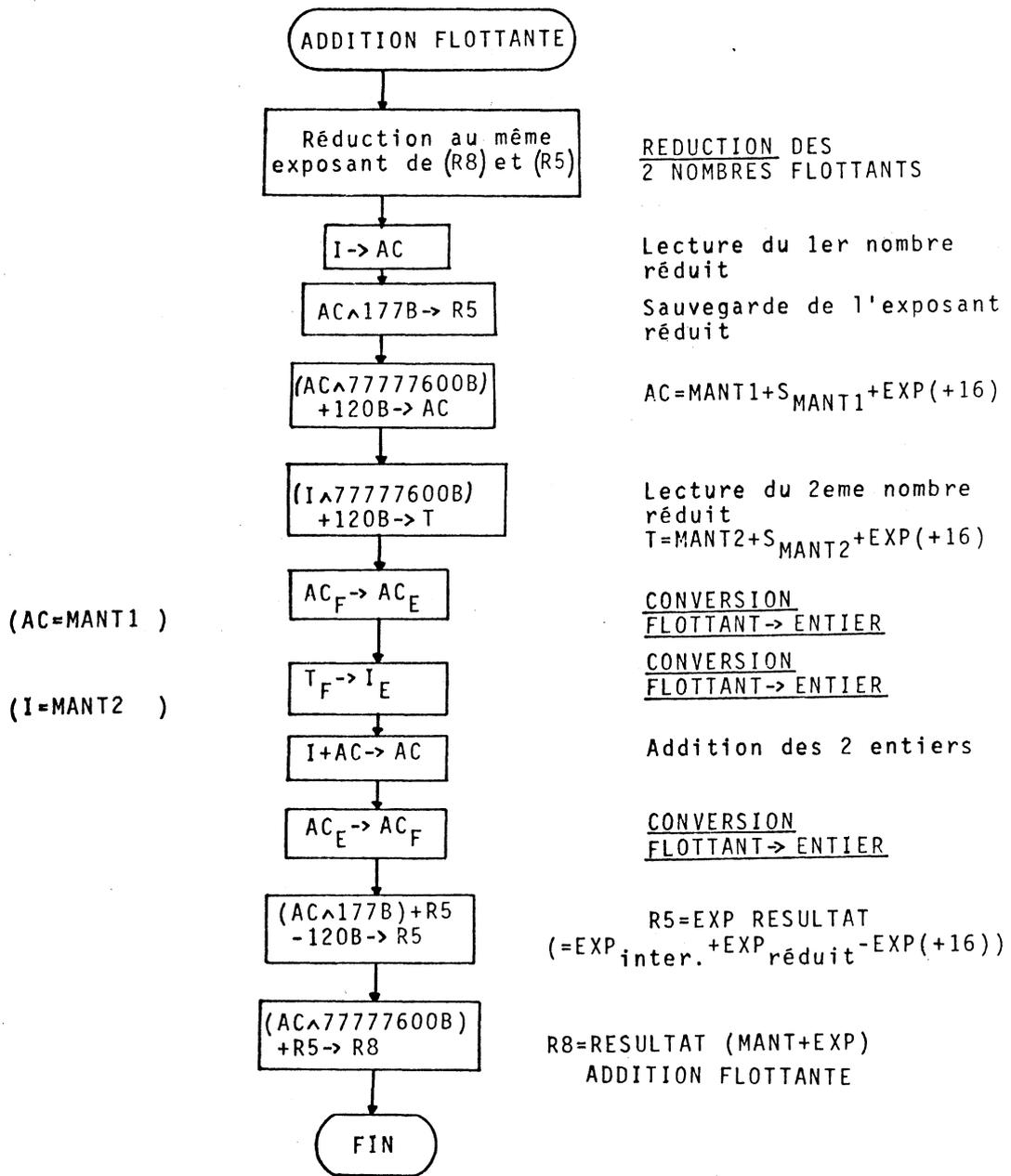


Figure 28: Organigramme du microprogramme MASS





3.2. L'instruction HODO 2 : traitement des données hodoscopes

La figure 29 donne le chaînage des tables générées par l'instruction.
La figure 30 donne l'organigramme du programme.

Des traitements différents doivent être effectués selon le type de données hodoscopes :

- coupure NTCO sur la multiplicité des compteurs R,
- préparation du test VR_3R_4 en mémorisant les compteurs R_3 et R_4 touchés dans chaque sextant,
- préparation du test P_1P_2 en mémorisant les éventuelles coïncidences P_1P_2 , ou en reconstruisant ces coïncidences à l'aide des compteurs P_1 et P_2 touchés dans chaque sextant.

A ces traitements s'ajoutent :

- la détection des adresses interdites,
- la détection de fin de données hodoscopes car le nombre de données hodoscopes est variable et différent pour chaque événement,
- le contrôle de la croissance des adresses.

Comme les données hodoscopes représentent en moyenne 40 mots, un débranchement multiple séquentiel pour chaque donnée entraînerait un temps d'exécution beaucoup trop important.

A l'aide d'une table de correspondance, un code opération est associé à chaque donnée hodoscope, et les débranchements multiples sont parallèles grâce aux codes de sauts conditionnés au code opération du contrôleur MCU :

- les 4 bits de poids fort définissent le type de la donnée hodoscope et le saut JPX exécute le branchement vers le traitement associé;
- le saut JPR conditionné par les 4 bits de poids faible permet un deuxième débranchement à l'intérieur d'une branche :
 - séparation R_1R_2 et R_3R_4 dans la branche R,
 - mise à 1 du bit correspondant à la valeur p_j d'un compteur P_j dans la branche P_1/P_2 .
- lorsque le traitement nécessite une mémorisation (P_1/P_2 , COINC P_1P_2 , R_3 , R_4), l'adresse mémoire, fonction du numéro de sextant, est fournie par les 16 bits de la zone complémentaire, ce qui évite un calcul d'adresse et une référence mémoire supplémentaire.

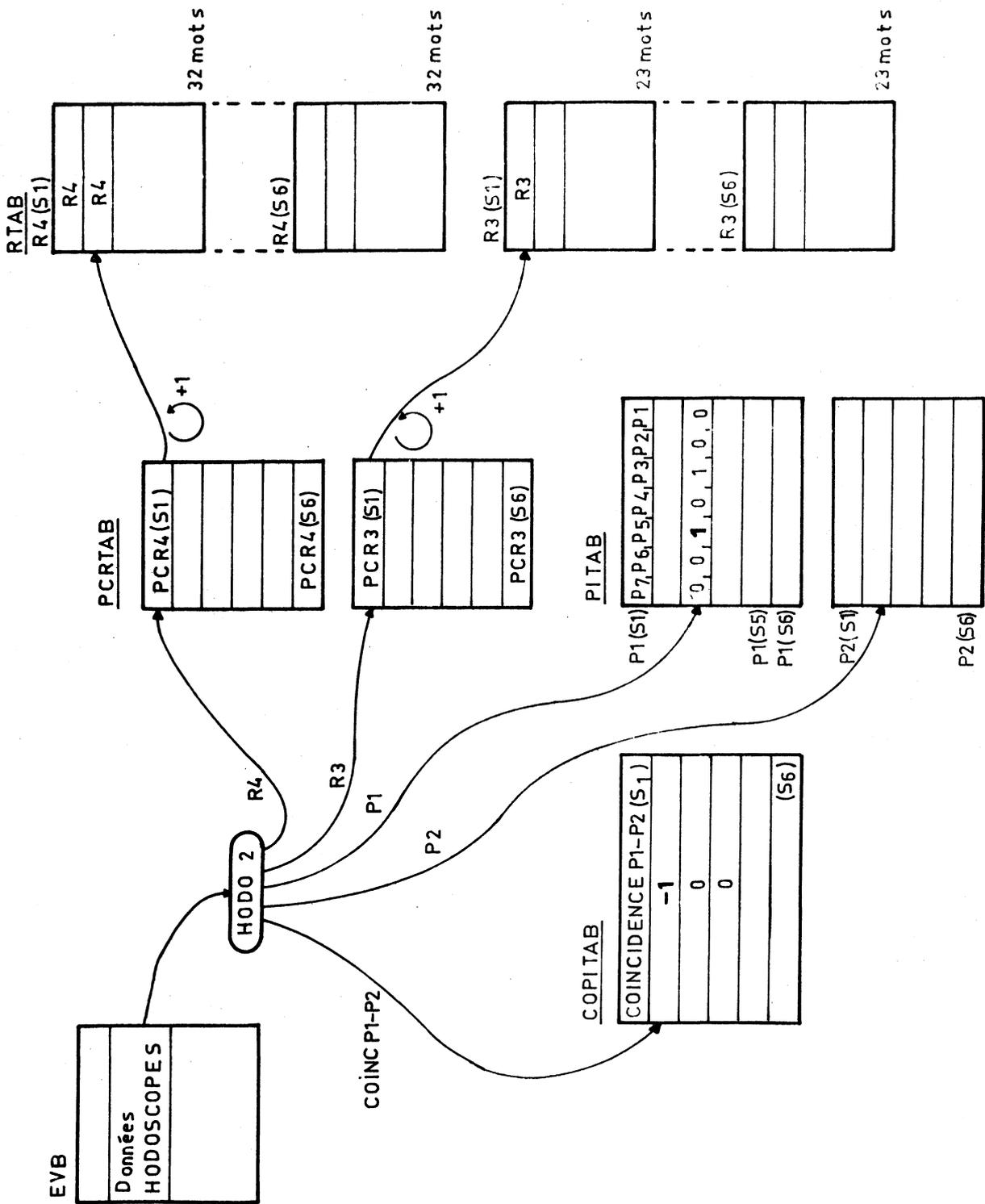


Figure 29 : Tables générées par l'instruction HODO 2

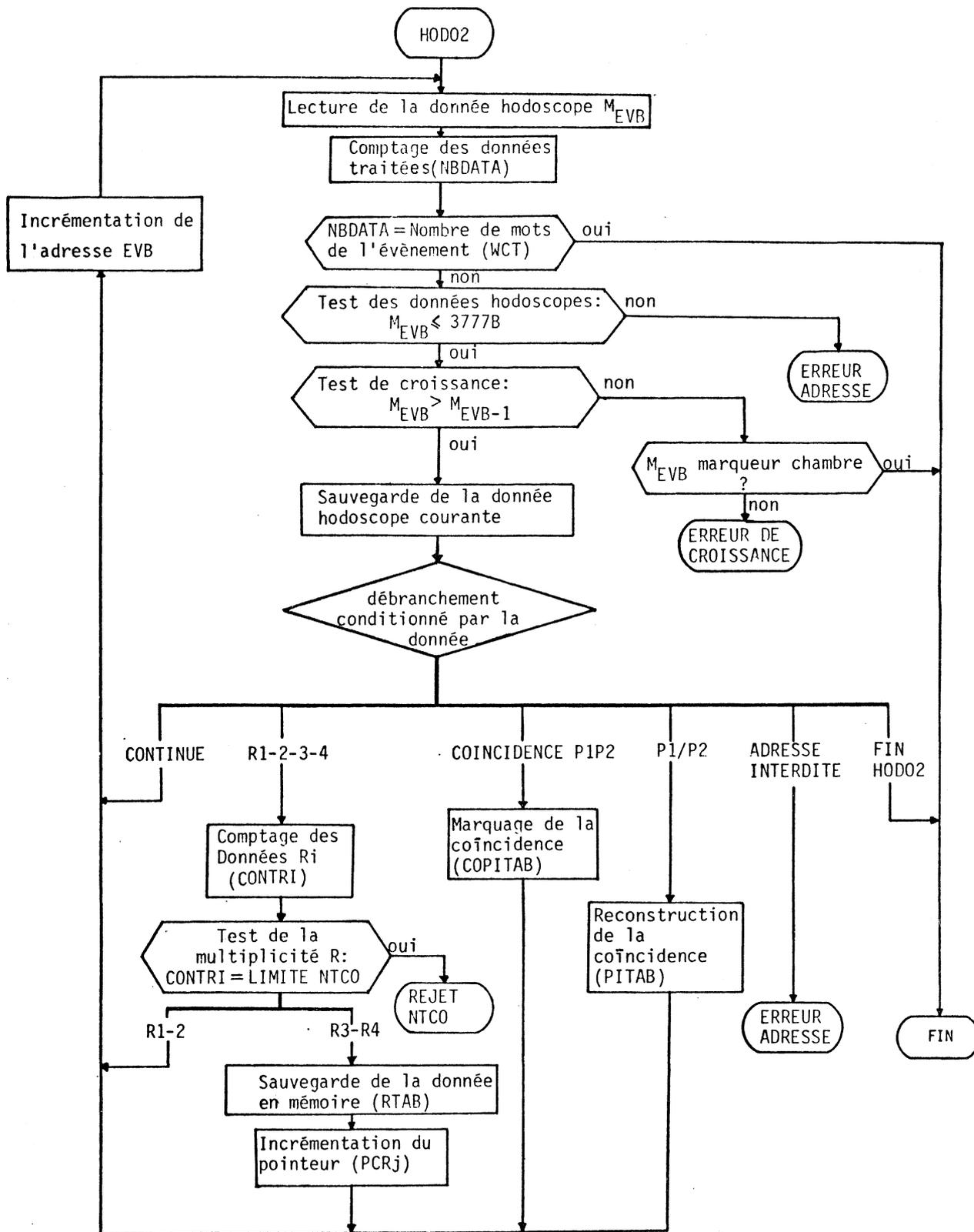


Figure 30-1: Organigramme de l'instruction HOD02

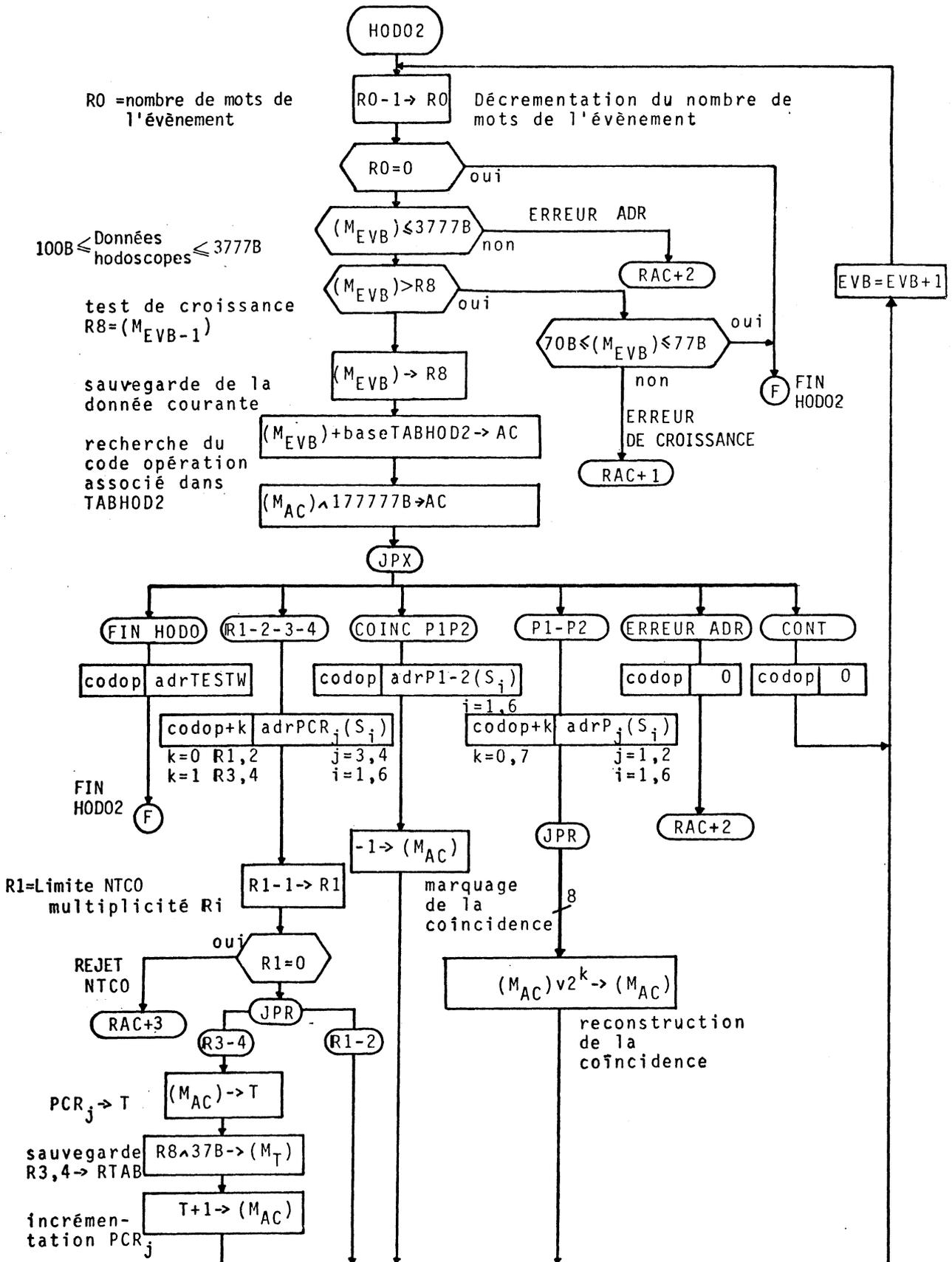


Figure 30-2: Organigramme du microprogramme HODO2

1. La branche R1-R2-R3-R4

Chaque donnée R_i décrémente le compteur NTCO qui fixe la limite de multiplicité des compteurs R_i dans un évènement. Si le compteur NTCO s'annule, ceci indique une multiplicité supérieure à la limite programmée et donne lieu à une sortie en rejet NTCO. Les données R3 et R4 sont mémorisées en fonction de leur numéro de sextant et le pointeur correspondant est incrémenté.

2. La branche COINC P1P2

Chaque coïncidence $P1 \times P2$ entraîne l'activation d'un marqueur, correspondant au sextant touché, dans la table de coïncidence COPITAB.

3. La branche P1/P2

La résolution en temps des circuits de coïncidence peut entraîner qu'une coïncidence $P1 \times P2$ n'ait pas été détectée alors que les compteurs P1 et P2 aient bien été touchés. En mettant à "1" le bit correspondant au numéro de compteur p_i touché pour chaque sextant des hodoscopes azimutaux P1 et P2, on reconstruit les coïncidences qui seront testées par le traitement P1P2.

4. La branche CONT

Certaines données hodoscopes ne donnent lieu à aucun traitement - c'est le cas des compteurs V notamment - : seul le compteur du nombre de mots de l'évènement est décrémente.

5. La branche ERREUR ADR

Lorsqu'une adresse-compteur interdite est détectée dans les données hodoscopes, le traitement s'achève en rejet ERREUR ADR.

6. La branche FIN HODO

Lorsqu'un marqueur de début de branche chambre (70-77B) est détecté, ou que le compteur du nombre de mots de l'évènement s'annule, le traitement des données hodoscopes est terminé et le programme se poursuit par l'exécution du test P1P2.

7. Le test de croissance:

Lorsque la donnée hodoscope courante n'est pas croissante par rapport à la donnée précédente et que ce n'est pas un marqueur de début de branche chambre, le traitement s'achève en erreur INCREMENT.

L'utilisation des codes JPX, JPR, et d'une table de correspondance en mémoire permet une grande souplesse de programmation:

- plusieurs traitements peuvent être ajoutés sans modification du microprogramme existant et sans perte de vitesse d'exécution, puisqu'aucune séquence de débranchement supplémentaire n'est nécessaire;
- le traitement de chaque donnée hodoscope peut être supprimé en remplaçant dans la table de correspondance, le code opération associé par le code CONTINUE. On peut ainsi, éviter les rejets systématiques lorsqu'un compteur tombe en panne et poursuivre quasi-normalement l'acquisition de données.

Le temps d'exécution de l'instruction est fonction du nombre et de la nature des données à traiter. Le traitement de chaque donnée nécessite:

2.05 μ s + :

0.40 μ s pour la branche CONT,

0.65 μ s pour la branche R1R2,

1.70 μ s pour la branche R3R4,

0.70 μ s pour la branche COINC P1P2,

0.65 μ s pour la branche P1P2,

0.85 μ s pour la branche ERREUR ADR,

ou 0.85 μ s pour la sortie en branche ERREUR DE CROISSANCE.

(Les organigrammes des traitements microprogrammés P1P2 et VR3R4 sont donnés en Annexe D.)

III. GENERATION DU LOGICIEL GESPRO

La figure 31-1 illustre les différentes étapes de génération du logiciel GESPRO.

1. Microprogramme

Le microprogramme-objet en langage machine peut être généré :

- par compilateur de microcode ^[10] exécuté sous système NORD, qui assemble le microprogramme source (en langage symbolique) et assure la tâche délicate de l'implantation en mémoire de microprogramme, en cherchant à optimiser l'occupation mémoire par un essai systématique de réutilisations des microinstructions déjà implantées.
- directement par le programmeur qui assure alors l'implantation en mémoire de microprogramme. Un programme de contrôle permet, dans ce cas, de vérifier qu'il n'y a pas d'erreurs de formats ou qu'une adresse mémoire de microprogramme n'a pas été allouée à deux microinstructions différentes.

2. Programme

Un assembleur GESPRO ^[11] accepte en un passage les programmes-source en langage symbolique et les traduit sous forme de programme-objet en langage machine.

Un chargeur GESPRO ^[11] translatable permet de charger ce programme-objet en mémoire GESPRO à une adresse paramétrable et de générer simultanément le fichier image mémoire correspondant. L'assembleur, l'éditeur de liens et le chargeur peuvent utiliser des jeux d'instructions quelconques et une syntaxe conversationnelle permet de modifier les fichiers des codes opérations associées.

3. Génération des fichiers-image

Un programme FORTRAN génère à partir des fichiers-objet, les fichiers-image de la mémoire de microprogramme et de la mémoire centrale GESPRO. Ces fichiers sont appelés par le programme d'acquisition pour charger les mémoires des calculateurs à l'initialisation du système.

Différentes options permettent de changer rapidement le jeu d'instructions, le programme GESPRO ou une table de constantes sans être obligé de régénérer toute l'image-mémoire.

La figure 31-2 donne un extrait du programme de filtrage-objet GESPRO.

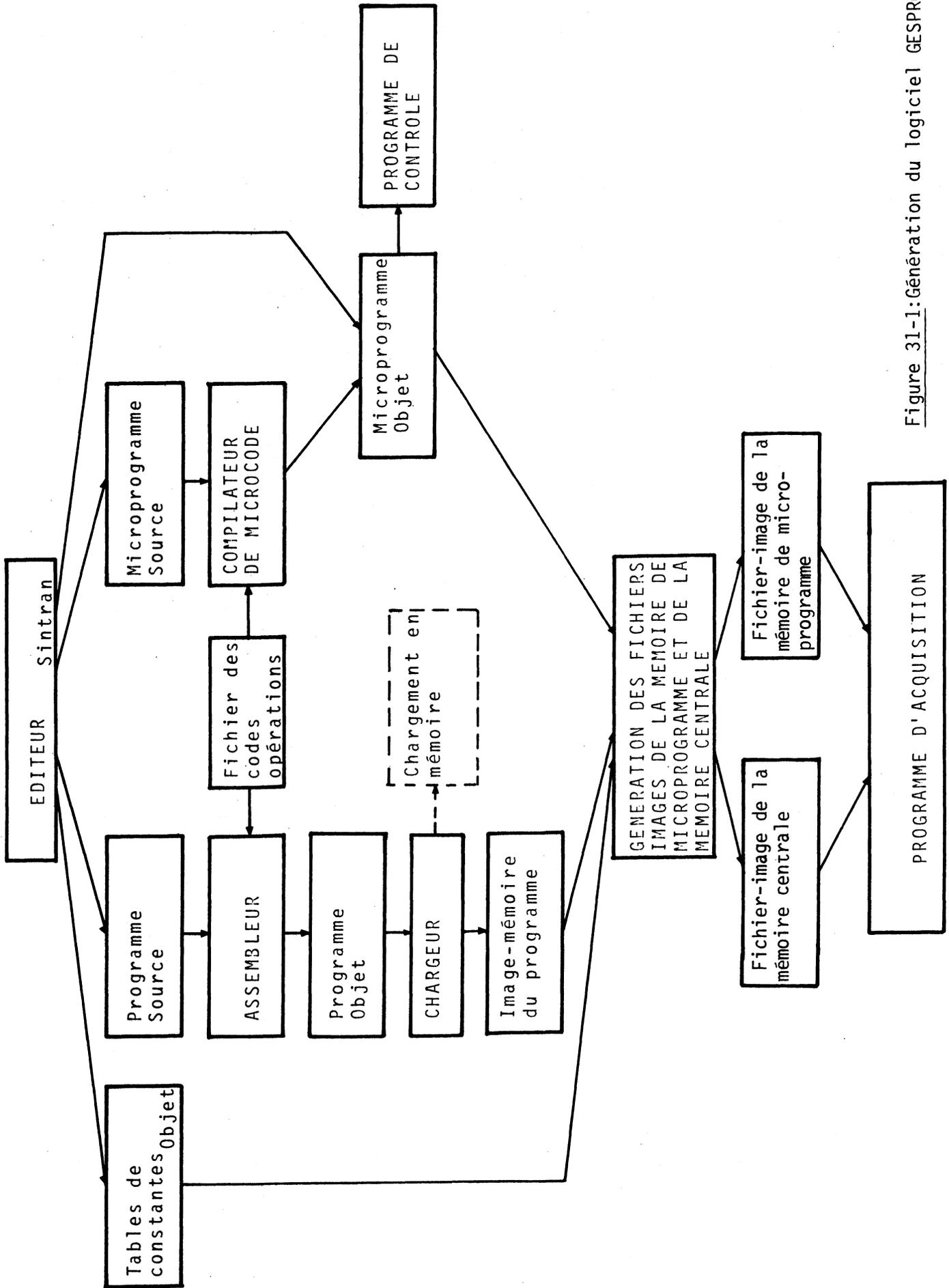


Figure 31-1: Génération du logiciel GESPRO

IV. LES PERFORMANCES INTRINSEQUES DU CALCULATEUR

1. Le temps d'exécution des instructions spécialisées

Le tableau ci-dessous résume les temps d'exécution des différentes instructions spécifiques au programme de filtrage. Le temps d'exécution des instructions qui dépendent de la structure particulière des événements a été calculé sur un événement moyen contenant:

- 2 sextants touchés
- 30 données hodosopes (dont 8 R1,R2 et 8 R3,R4)

SDI	4.0 μ s
LDPT	8.5 μ s
MASS	37.8 μ s
CPMASS	2.5 μ s
STA	2.8 μ s
HOD01	5.1 μ s
WCT	2.3 μ s
HOD02	83.2 μ s
(+P1P2)	4.5 μ s
VR3R4	30.1 μ s
GOOD/BAD	1.7 μ s

2. Apport des opérateurs cablés et de la microprogrammation

Pour déterminer l'apport des opérateurs cablés et de la microprogrammation, les traitements de filtrage ont été écrits en mode programmé, à partir des instructions banalisées du jeu d'instructions GESPRO.

Les temps d'exécution et le nombre de références mémoire pour chaque mode de traitement sont regroupés dans le tableau suivant:

TRAITEMENT	MODE	TEMPS D'EXECUTION (µs)	Nombre de REFERENCES MEMOIRE	Nombre de FETCH PHASE	GAIN en Temps d'exécution		% du temps d'exécution: REFERENCES MEMOIRE FETCH PHASES supplémentaires à celles décrites par l'algorithme		REMARQUES -t _{cycle horloge} = 150 ns -t _{cycle mémoire} = 400 ns
SDI	µP	5.2	9	1	5	23 %	27 %	Cas le plus défavorable: OVf Prog: avec sauvegarde et changement de contexte	
	Prog	26	21	11					
RECHERCHE DES PLUS MOMENTS TRANSVERSE	Cablé	4.8	0	0	31.2	130	1 %	50 %	
	µP	150	204	1	4.2				
	Prog	624	221	18					
MULTIPLICATION FLOTTANTE	Cablé	0.2	0	0	255	4200	22 %	15 %	
	µP	51	22	1	16.5				
	Prog	840	391	187					
GENERATION DE TABVAL	µP	28	50	0	6.9	23 %	65 %	6 sextants touchés	
	Prog	191	140	177					
M ² /2 (avec multiplieur)	µP	13	7	0	3.3	24 %	47 %		
	Prog	43	28	29					
MASS(2 sextants)	µP	37.8	34	1	4.5	21 %	57 %		
	Prog	170	107	140					
MASS(6 sextants)	µP	317	209	1	3.4	20 %	49 %		
	Prog	1076	647	753					
CPMASS	µP	3.4	2	1	6.4	9 %	41 %	Cas le plus défavorable: 2 flottants égaux	
	Prog	22	6	14					
STA	µP	2.8	4	1	2.9	12 %	26 %		
	Prog	8	6	4					
HOD01	µP	5.1	7	1	6.5	27 %	51 %	Evènement moyen	
	Prog	33	25	25					
WCT	µP	2.0	1	1	3.2	15 %	54 %		
	Prog	6.5	3	6					
HOD02	µP	83.2	100	1	7.8	22 %	63 %	Evènement moyen	
	Prog	651	386	586					
P1P2	µP	4.5	7	0	4.8	0 %	51 %	1 coincidence + 1 reconstruction	
	Prog	22	7	17					
GOOD/BAD	µP	1.7	4	1	3.5	33 %	23 %		
	Prog	6	6	3					

Ce tableau montre clairement l'apport des opérateurs câblés au niveau des temps d'exécution, mais l'importance de cet apport dépend :

- du degré de parallélisme de la structure de l'opérateur,
- de l'importance que conserve le mode séquentiel dans l'exécution du traitement,
- de la structure de la liaison opérateur-unité centrale (l'interconnexion à travers des registres-tampons permet de masquer le temps d'exécution de l'opérateur par une activation en simultanéité avec d'autres tâches du traitement).

La comparaison entre les modes programmés et microprogrammés utilisant les mêmes opérateurs met en évidence deux apports principaux de la microprogrammation :

- disparition des phases de recherche du code opération et des références mémoires associées, qui représentent, en moyenne plus de 50 % du temps d'exécution du traitement programmé.
- disparition de la quasi-totalité des sauvegardes temporaires de variables en mémoire centrale par la gestion rationnelle de tous les registres internes. L'ensemble de ces références-mémoire supplémentaires à celles décrites par l'algorithme représente en moyenne 22 % du temps du traitement programmé.

Le gain moyen introduit par la microprogrammation sur le temps d'exécution est environ de 4,5 pour le calculateur GESPRO NA10, mais il convient de faire les remarques suivantes :

- Les temps d'exécution de la plupart des traitements en mode programmé GESPRO sont déjà des résultats performants comparés à ceux des minior-dinateurs banalisés.
- Les instructions courantes du jeu d'instructions GESPRO, sont microprogrammées, donc optimisées en temps d'exécution.
- Ce gain est fortement dépendant de la structure matérielle du calculateur. Programme et microprogramme disposent de la même puissance matérielle :
 - machine 24 bits,
 - structure multi-busses,
 - opérateurs spécialisés,
 - module d'entrées/sorties spécifiques à l'environnement.

Ce masquage du gain logiciel par le gain matériel, se traduit par la diminution du gain programmé/microprogrammé dès que la puissance matérielle mise en oeuvre augmente.

Ainsi, la multiplication flottante, sans opérateurs spécialisés est 16 fois plus rapide en mode microprogrammé qu'en mode programmé. De même, la génération de TABVAL, qui manipule des tableaux de variables en mémoire est 7 fois plus rapide en mode microprogrammé, alors que le gain du calcul de masse n'est que de 3.3 car les opérations de multiplication et d'addition flottantes font appel au multiplieur et restent microprogrammées dans les deux versions, programmée et microprogrammée.

Lorsque l'algorithme comporte de nombreux débranchements (HOD01, HOD02,

CPMASS,...) le gain introduit par la microprogrammation est plus important que pour les algorithmes ne contenant que des opérations élémentaires (GOOD, BAD, WCT,..) parfaitement décrites par les instructions banalisées du jeu d'instructions.

V. LES RESULTATS EXPERIMENTAUX

Les millions d'événements acquis par le système de filtrage complet (4 couples GESPRO-EVB) au cours des deux premiers trimestres 1982, ont permis de dégager les résultats suivants :

1. Les temps de filtrage

Les temps de filtrage correspondent aux intervalles de temps qui séparent l'acquiescement du signal STA, d'une réponse GOOD ou BAD, pour les branches successives de rejet du programme de filtrage, décrit précédemment.

Ce temps tient compte des temps de traitements des différentes coupures traversées, mais aussi du temps nécessaire à la prise en compte du signal STA et du temps d'émission de la réponse GOOD/BAD.

	BIAISE	NON BIAISE
STA → IRAWMAX	90 μ s	90 μ s
STA → NTCO (10)	170 μ s	170 μ s
STA → NTCO (20)	200 μ s	200 μ s
STA → NTCO (40)	270 μ s	270 μ s
STA → TRIMUON	280 μ s	-
STA → NBSEXT < 2	285 μ s	-
STA → LIMMAS	295 μ s	280 μ s
STA → PSI	300 μ s	285 μ s
STA → P1P2	310 μ s	295 μ s
STA → VR3R4	350 μ s	330 μ s
STA → GOOD (aucun rejet)	370 μ s	350 μ s

2.Exemple d'acquisition à haute intensité ($I=2 \cdot 10^9 \pi/c$)

RUN 1973

Beam particule: π at 200 Gev/c

Trisser type: Masic box

Masic box trisser: dimuon

Prescaling of J/Psi: 1

Sampling rate of unbiased trissters: 15

4 Event Buffers enabled: 0, 1, 2, 3

Tests done by Gespro:

Test 1: Trimuon mode: flas

Test 2: Mass calculation mode: flas

Test 3: Mass in J/Psi region mode: flas

-prescaling region: 2000-4000 Mev

-prescaling factor: 1

Test 4: Cut on nb. of raw data words mode: rejection

-Max. nb. of raw data words: 800

Test 5: Cut on nb. of hodoscopes words mode: rejection

-Max. nb. of hodoscopes words: 40

Test 6: P1-P2 not valid non active

Test 7: U-R3-R4 not valid mode: flas

RUN 1973	17:14 243 BURSTS	15/06/1982	20061 EVENTS		
GESPRO STATUS					
Gespro number	0	1	2	3	SUM
Status	run	run	run	run	
SDI	94568	9693	637	49	104947
STA	32386	3279	206	13	35884
GOOD total	18161	1797	96	7	20061
BAD	14225	1482	110	6	15823
DIE(data input error)	0	0	0	0	0
T.O. (acquisition)	0	0	0	0	0
OVF(buffer overflow)	62182	6414	430	35	69061
STOP	0	0	0	0	0
GOOD Unbiased	2156	198	8	0	2362
RMH Error Gespro	1	0	0	0	1
Norm. J/Psi	146	14	1	0	161
TESTS(flas + reject)	000137	000137	000137	000137	
REJECTIONS	000030	000030	000030	000030	

RUN 1973 LOG:	FLAGGING	REJECTION	TIJ:	FLAGGING	REJECTION
TRIMUON	45	0	12	87	184
PSI	9288	0	34	125	385
MASS<L	616	0	56	140	300
IRAWMAX	0	11148	13	71	255
NTCO	0	4675	15	137	616
P1F2	0	0	24	124	370
VR3R4	2047	0	26	74	124
SIGN	0	0	35	128	469
TO EVB	0	0	46	110	216
ADR ERROR	0	0	16	123	333
67B ERROR	0	0	36	146	510
INCR ERROR	0	0	23	179	422
-----	0	0	25	300	1326
WC ERROR	1	0	45	670	8674
NBSEXT<2	1666	0	14	245	1508
UNBIAS	2362	0	00	1	171

Le STATUS GESPRO est disponible pendant l'acquisition et permet de contrôler le bon transfert des événements grâce aux relations de fermeture

.SDI = STA + DIE + OVF + TO (à 1 près par calculateur, si le transfert RMH du dernier événement alloué a été interrompu, le test du registre d'états n'étant exécuté qu'à l'événement suivant)

STA = GOOD + BAD + STOP

De même, un contrôle des rejections est fourni par les rapports :

$\frac{SDI}{OVF}$ = Réjection Memory OVERFLOW (EVB)

$\frac{STA - GOOD \text{ Unbiased} - RMH \text{ Error}}{GOOD - GOOD \text{ Unbiased} - RMH \text{ Error}}$ = Réjection globale GESPRO

$\frac{SDI - GOOD \text{ Unbiased} - RMH \text{ Error}}{GOOD - GOOD \text{ Unbiased} - RMH \text{ Error}}$ = Réjection globale GESPRO + EVB

Sur les 104947 événements signés par la sélection câblée (SDI) :

- 69061 événements supérieurs à 1k mots ont été rejetés par la coupure OVF (65.8 %)
- aucune erreur ou dépassement ne se sont produits au cours des transferts RMH-EVB (DIE = 0, TO = 0)
- 35884 événements ont été analysés par le programme de filtrage après STA.

Sur ces 35884 événements :

- 15823 événements ont été rejetés (15 %) :
 - 11148 événements par la coupure IRWMAX (10.6 %)
 - 4675 événements par la coupure NTCO (4,4 %)
- 20061 événements ont été acquis (19,1 %), dont :
 - 2362 événements non biaisés (2,2 %)
 - 2047 événements "rejetés" par le test VR3R4 (1,9 %)
 - 1 erreur "RMH" : Word Count = < 56B

Parmi les 15651 événements ayant satisfait à tous les tests (14,9 %) :

- 1666 événements avaient moins de 2 sextants touchés (1,6 %)
- 45 événements étaient des trimuons (4,2 ‰)
- 619 événements avaient une masse < 2 GeV/c (5,8 ‰)
- 9288 événements avaient une masse comprise entre 2 et 4 GeV (8,8 ‰)

En appliquant tous les tests en mode rejet(y compris NBSEXT<2 et la coupure en masse)on rejeterait 93.8 % des évènements(soit une réjection égale à 25).

Le nombre d'évènements alloués à chaque couple GESPRO-EVB, montre que pour une énergie du faisceau de Π de 200 GeV, le système de filtrage est loin de la saturation, puisque le 4e GESPRO ne traite que 5‰ des évènements

signés par le déclenchement câblé, et qu'on admet que le système atteint la saturation lorsque chaque GESPRO traite 90 % de l'information traitée par le GESPRO qui le précède, dans la priorité des allocations.

3. Réjections des différentes coupures

Les taux de réjection présentés dans le tableau ci-dessous ont été mesurés pour une acquisition à intensité moyenne ($I \approx 1.4 \times 10^9 \pi/c$), et des limites IRAWMAX = 800 et NTCO = 40.

Type de Rejet	Réjection
Dépassement de capacité mémoire MEMORY OVERFLOW	1.65
Nombre de mots de l'événement IRAWMAX (= 800)	2.37
Multiplicité R1-2-3-4 NTCO (= 40)	1.20
Coïncidence P1-P2	1.30
Alignement VR3R4	1.10
Coupure en masse	2.4
OVF + IRAWMAX + NTCO	4.75

4. Le spectre de masse

Le spectre des masses calculées en ligne par GESPRO (figure 32-1) correspond aux résultats escomptés par la simulation.

La résolution de la masse calculée par GESPRO est limitée par la résolution des compteurs hodosopes. La comparaison avec la masse calculée hors ligne à partir des données des chambres (figure 32-2) montre que la résolution est suffisamment bonne ($\approx 11\%$ pour la limite à 4 GeV) pour permettre d'appliquer la coupure en masse sans introduire une inefficacité importante ($< 1.2\%$ pour les événements de masse ≥ 4 GeV).

nombre d'évènements

- 98 -

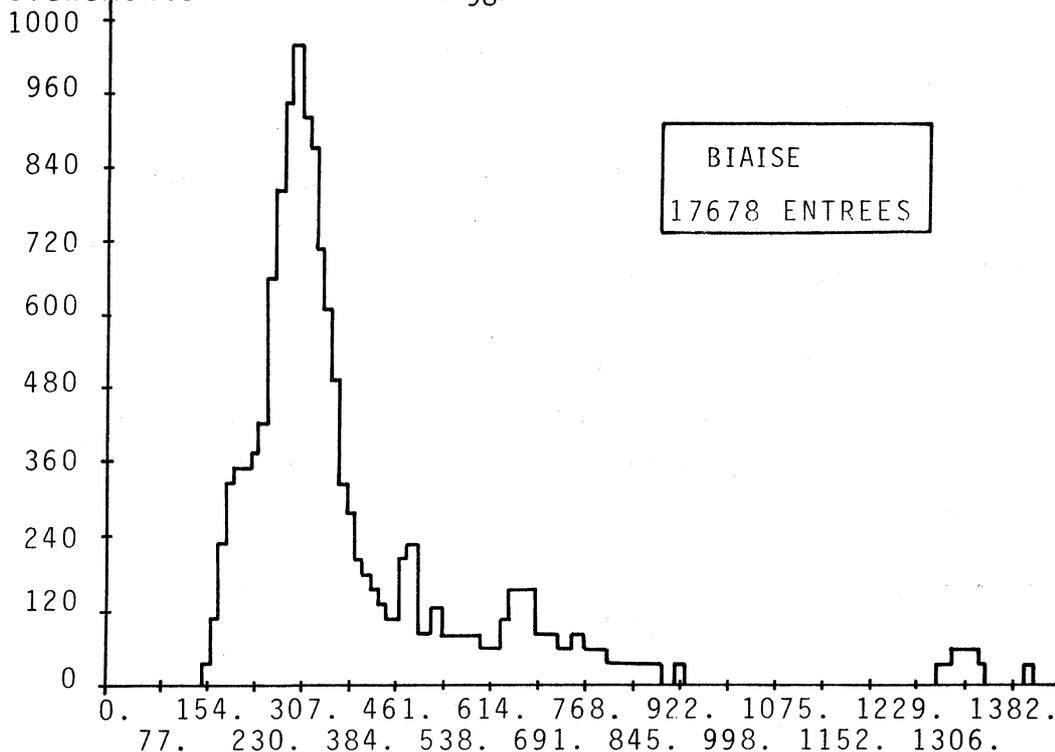


Figure 32 -1. Histogramme du nombre d'évènements biaisés en fonction de la masse du dimuon

Masse (MeV)

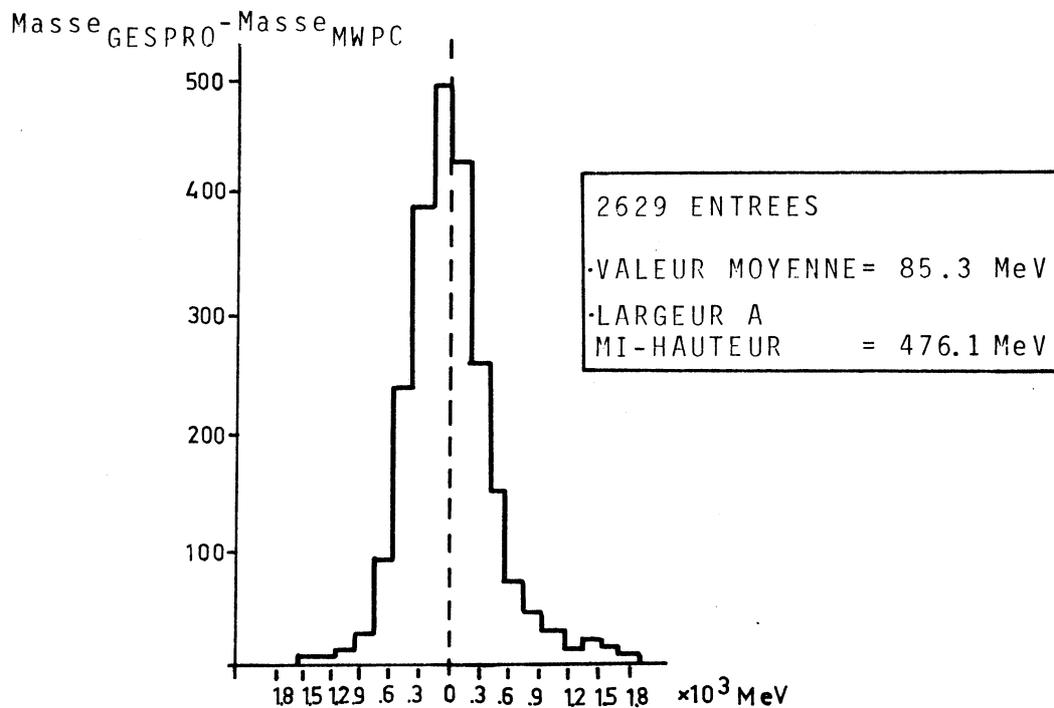


Figure 32 -2. Résolution en masse calculée en ligne par GESPRO et la masse calculée hors ligne à partir des données chambres.

5. Variation de la réjection en fonction de l'intensité

Lorsque l'intensité du faisceau croît, le nombre d'accidentelles augmente, et le nombre de déclenchements par cycle s'élève comme le montre la courbe 1 figure 33.

La réjection du système augmente simultanément (courbe 2 figure 33), mais son accroissement est moins rapide que le nombre de déclenchements : le taux de fortuites acquies par le système augmente avec l'intensité du faisceau.

6. Variation du temps actif en fonction de l'intensité

La diminution de la réjection se traduit par une diminution du temps actif (fig.33-3). Cette diminution est due au temps mort introduit par les temps de transferts RMH-EVB.

Pour réduire ce temps mort, deux améliorations peuvent être apportées au programme de filtrage :

- La coupure en masse est appliquée comme déclenchement logiciel et rejette les événements de faible masse et de masse intermédiaire qui représentent 50 % des événements signés par le déclenchement câblé, en un temps moyen de décision de 31.5 μ s (histogramme figure 34).

Le temps moyen du transfert RMH-EVB étant de 150 μ s, l'introduction de ce déclenchement permet de réduire le temps de transfert global de 30 %.

- Le traitement des données hodoscopes débute dès que ces données sont mémorisées dans l'EVB, en vol de cycle avec le transfert RMH-EVB des données chambres, afin d'interrompre l'acquisition de l'événement le plus rapidement possible s'il ne satisfait pas aux critères du test IRAMAX ou NTCO.

7. Le contrôle de l'expérience

7.1. La surveillance de l'acquisition par le comptage des PSI

Le calcul de masse et la comparaison aux limites de masse, bien qu'inutilisés comme rejet dans le programme de filtrage actuel, se sont révélés particulièrement efficaces pour la surveillance de la qualité de l'acquisition, grâce au comptage des PSI.

Toute diminution importante du rapport nombre de PSI/nombre de particules incidentes signale un problème de fonctionnement qui n'est pas obligatoirement détecté par les autres contrôles de l'expérience, par exemple, une dérive en tension d'une alimentation haute tension,...

Nombre de déclenchements par cycle

- 100 -

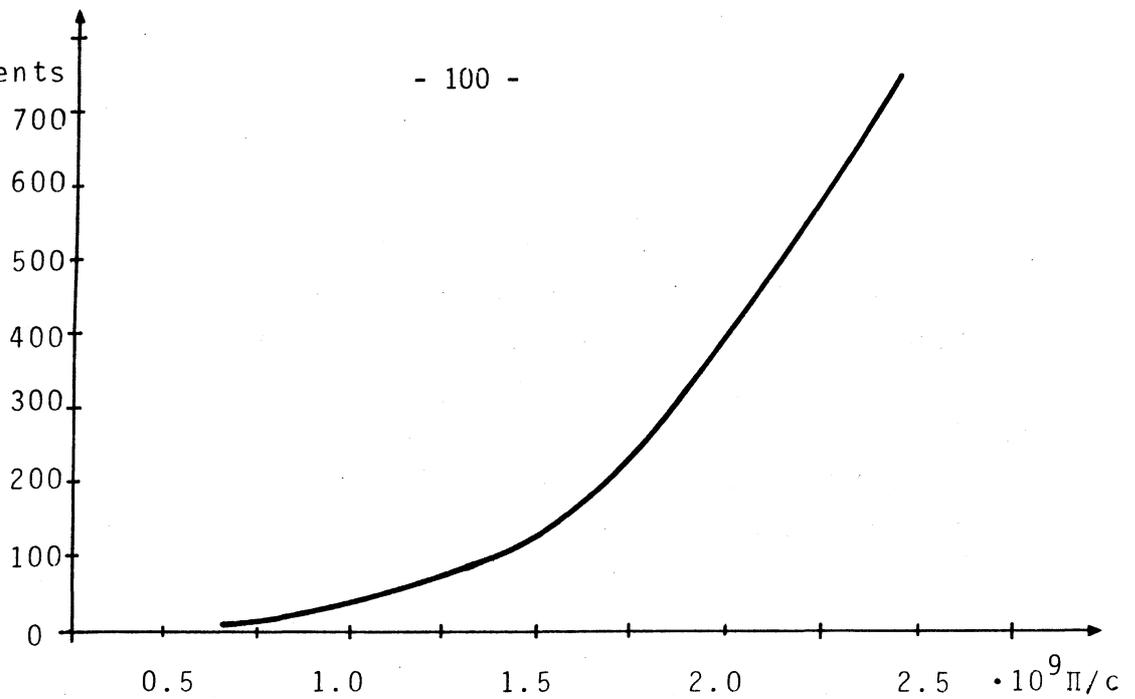


Figure 33.1. Variation du nombre de déclenchements par cycle en fonction de l'intensité

Réjection
Mémoire OVF
+IRAWMAX
+NTCO

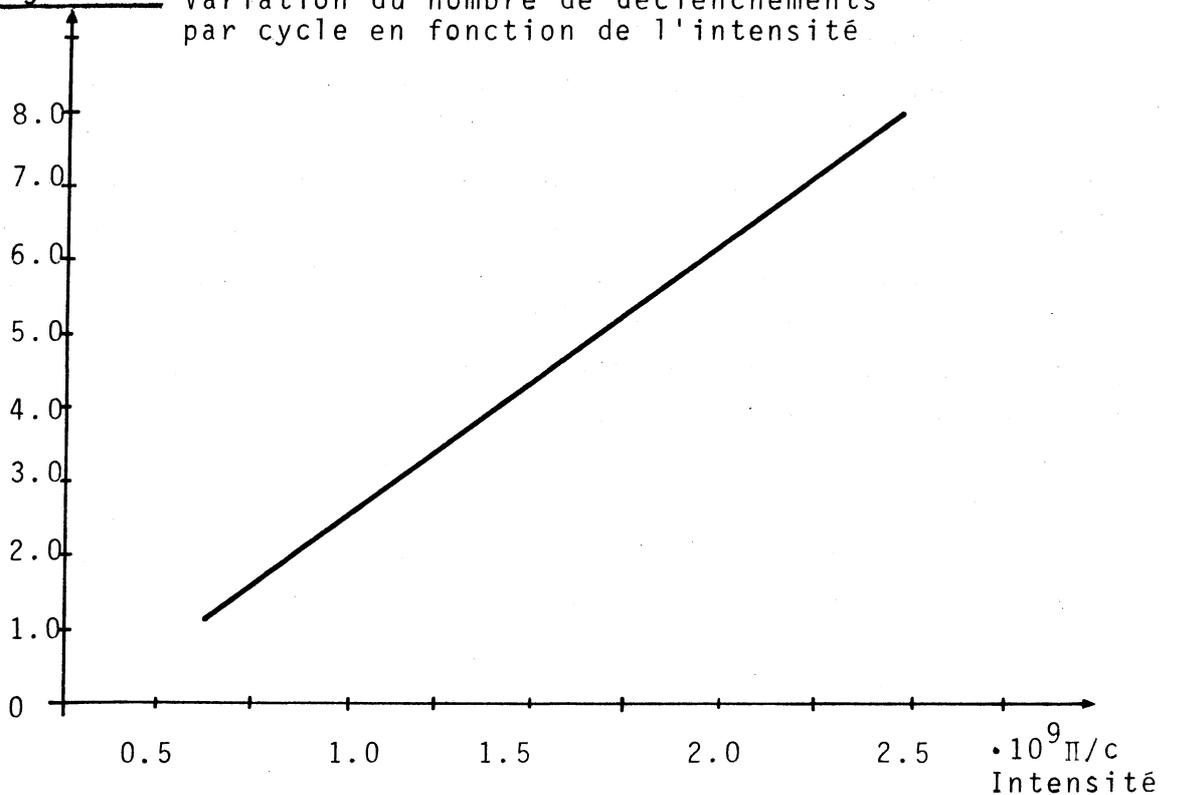


Figure 33.2. Variation de la réjection en fonction de l'intensité

Temps actif %

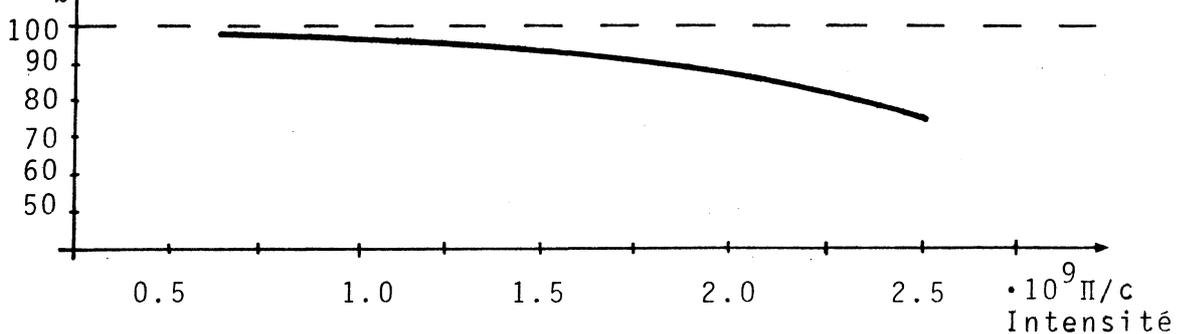
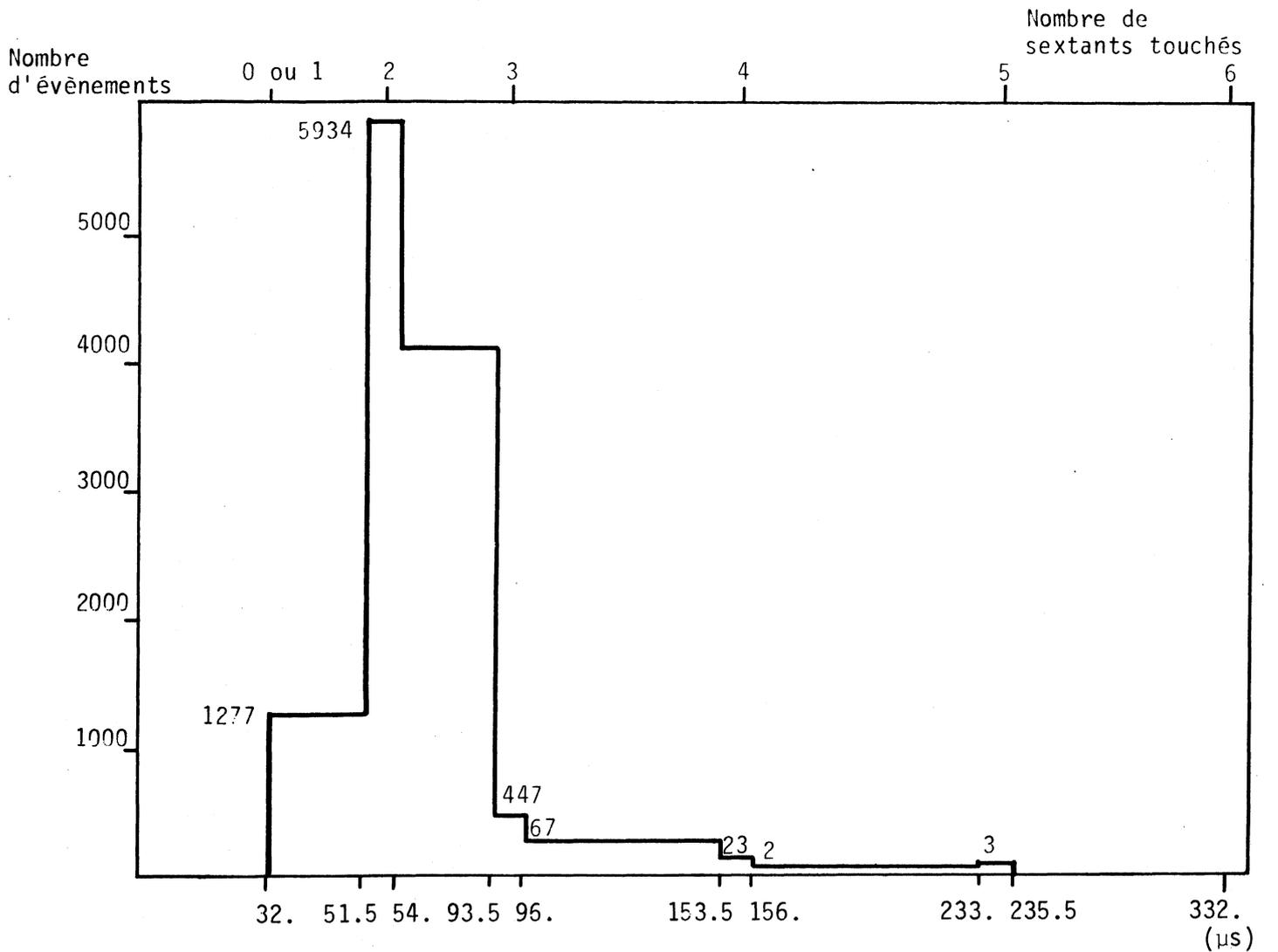


Figure 33-3. Variation du temps actif du système en fonction de l'intensité



Intensité: $1.4 \cdot 10^9 \pi/c$
11910 EVENEMENTS:
• 3932 $2\text{GeV} \leq \text{Masse} \leq 4\text{GeV}$
• 294 $\text{Masse} < 2\text{GeV}$
• 1277 $\text{NBSEXT} < 2$

temps de décision
(SDI+LDPT+MASS+CPMASS)

Figure 34 : Histogramme du nombre d'évènements signés par le déclenchement cablé en fonction du nombre de sextants touchés et temps de décision associé du déclenchement logiciel sur la coupure en masse

7.2. Test et maintenance du système d'acquisition

Pour perdre le moins de temps faisceau possible, tous les tests de fonctionnement du déclenchement et de l'acquisition en général, doivent pouvoir se faire sans faisceau.

Toute la logique est programmable, tous les tests du déclenchement, de vérification des canaux, du logiciel GESPRO et du fonctionnement de l'acquisition peuvent être exécutés en simulation.

Des programmes de tests effectués par GESPRO permettent de contrôler la logique du déclenchement et le bon fonctionnement des détecteurs très rapidement sans aucune modification matérielle du système d'acquisition.

7.3. Mesure de l'efficacité du déclenchement

L'efficacité du déclenchement câblé (compteurs R, matrice, logique intersextants) est un paramètre extrêmement important de l'expérience. Pour mesurer cette efficacité, on dispose d'un autre type de déclenchement n'utilisant que les compteurs des hodoscopes azimutaux P_1 et P_2 . Mais ce déclenchement, très sensible au halo, est peu sélectif, et entraîne un taux de déclenchement de plusieurs milliers d'événements par cycle-faisceau (3000 à 4000), pour un rendement à la reconstruction inférieur au pour cent.

Un programme GESPRO, contenant de nouvelles instructions microprogrammées de reconstructions de traces, est en cours de développement. En imposant des contraintes supplémentaires d'alignement sur les données chambres, il réduit le taux de déclenchement d'un facteur 5 et accroît le rendement à la reconstruction du même facteur.

• Deux résultats importants de l'acquisition illustrent les performances et la qualité du système de filtrage GESPRO-EVB:

A $2 \cdot 10^9$ II/cycle:

- le temps actif du système d'acquisition est de 90% ;
- 80% des événements acquis sont restructuribles (contre 7% sans filtrage GESPRO-EVB).

C O N C L U S I O N

L'introduction d'un ensemble de filtrage microprogrammé dans la chaîne d'acquisition de l'expérience NA10 a permis:

- une sélectivité accrue du déclenchement,
- une diminution du temps mort de l'acquisition,
- un contrôle plus strict du fonctionnement de l'expérience.

Les différentes tâches exécutées par GESPRO témoignent de ses capacités d'adaptation: processeur de déclenchement, de filtrage et de contrôle.

Les opérateurs cablés intervenant dans le système ont été définis lors de la phase de conception. Ils correspondaient, alors, aux besoins d'un traitement bien défini à cet instant. Le temps de réalisation de tels opérateurs, rend difficile leur introduction en cours d'expérience.

Le programme de filtrage a évolué, parce que les microprocesseurs avaient accès à la totalité de l'information, et parce que la microprogrammation est suffisamment performante et souple, pour permettre l'écriture de nouveaux traitements.

La nouvelle génération d'expériences prévues dans le domaine de la physique des Hautes Energies verra une augmentation dans la taille, mais aussi dans la complexité des expériences. Le nombre et la complexité des détecteurs mis en oeuvre entraînera un accroissement:

- du volume de l'information à traiter,
- de la difficulté du contrôle des expériences.

Au niveau du déclenchement, la logique cablée subsistera, mais un contrôle programmable sera indispensable. La diversité des déclenchements nécessitera des filtres raffinés permettant, par exemple, de corrélérer l'information des différents détecteurs.

Au niveau de l'acquisition, la quantité d'information à acquérir sera, non seulement volumineuse, mais également plus dispersée. La tendance actuelle, consistant à introduire des microprocesseurs au niveau de la lecture de l'information, va s'affirmer. En effet, la distribution de l'intelligence permet d'atteindre un grand degré de parallélisme, indispensable:

- à la réduction des temps de transferts,
- à l'allongement des temps de traitements susceptibles de réduire les temps de transferts par leur sélectivité.

La réussite de ces projets nécessitera de lourds investissements dans le matériel et le logiciel, et une bonne maîtrise des techniques décrites dans ce mémoire.

ANNEXE A

MULTIPLICATION FLOTTANTE :

PROGRAMMEE

MICROPROGRAMMEE

CABLEE

MULTIPLICATION FLOTTANTE PROGRAMMEE

(24 X 24 bits)

Test des opérandes

```
LD,A FLOT1
JZRO,A FIN          FLOT1=0 ->fin de la multiplication
LD,B FLOT2
JZRO,B FIN          FLOT2=0 ->fin de la multiplication
```

Traitement des exposants

```
ANDIM,A '177        A=EXP1
ANDIM,B '177        B=EXP2
RADD,B              A=EXP1+EXP2
SUBIM,A '64         Normalisation de l'exposant
ST,A EXP
CPONE EXP,'200      Test du bit 8
JMP OVF             Dépassement de capacité
```

Fractionnement des mantisses:16 bits -> 2 x 8 bits

```
LD,A FLOT1
SHR,A '10           Décalage 8 fois à droite
COPY,A,B
ADIM,B '277         Masquage des 8 bits MSB
ST,B LSB1
SHR,A '10           Décalage 8 fois à droite
ST,A MSB1
```

```
LD,A FLOT2
SHR,A '10           Décalage 8 fois à droite
COPY,A,B
ADIM,B '277         Masquage des 8 bits MSB
ST,B LSB2
SHR,A '10           Décalage 8 fois à droite
ST,A MSB2
```

Multiplication selon l'arbre de Wallace

```
LD,B MSB1
JMPM MULT           Appel sous-programme MULT
SHL,A '10           Résultat MSB1xMSB2 décalé 8 fois à gauche
ST,A RESULT4
```

```
LD,A LSB2
JMPM MULT           MSB1xMSB2
ST,A RESULT2
```

```
LD,B LSB1
LD,A MSB2
JMPM MULT           LSB1xMSB2
ST,A RESULT3
```

```
LD,A LSR2
JMPM MULT          LSB1xLSB2
SHR,A '10          Résultat décalé 8 fois à droite

ADD,A RESULT2     A+RESULT2->A
ADD,A RESULT3     A+RESULT3->A
ADD,A RESULT4     A+RESULT4->A
ANDIM,A '77777400 A=Mantisse résultat

Normalisation de la mantisse

COPY,A,B
ANDIM,B '40000000 Masque bit MSB
JNZR,B SUITO      Test bit MSB (=1 ->Suite)
SHL,A 1           Renormalisation:mantisse décalée 1 fois à
DCM EXP          exp-1->exp      gauche
JMP SUITO

SUITO OR,A EXP     Mantisse v Exp
      ST,A RESULTAT ->Résultat

FIN
```

Sous-programme MULT A x B -> A (8 x 8 bits)

```
Test des operandes

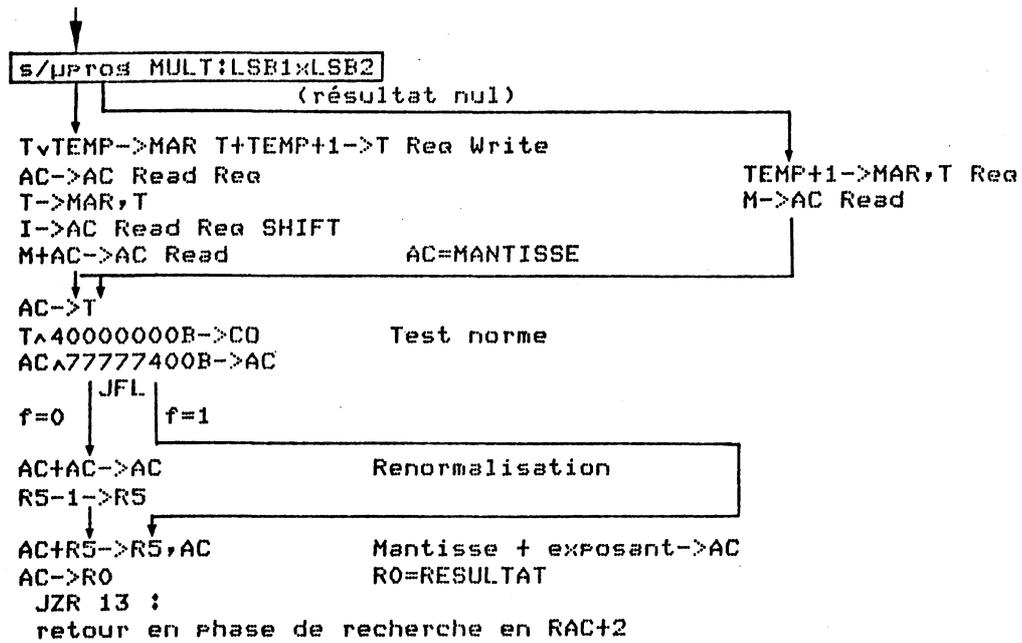
JZRO,A RET1       A=0 ->fin du sous-programme
JZRO,B RET1       B=0 ->fin du sous-programme

STIM RESULT,0    Initialisation

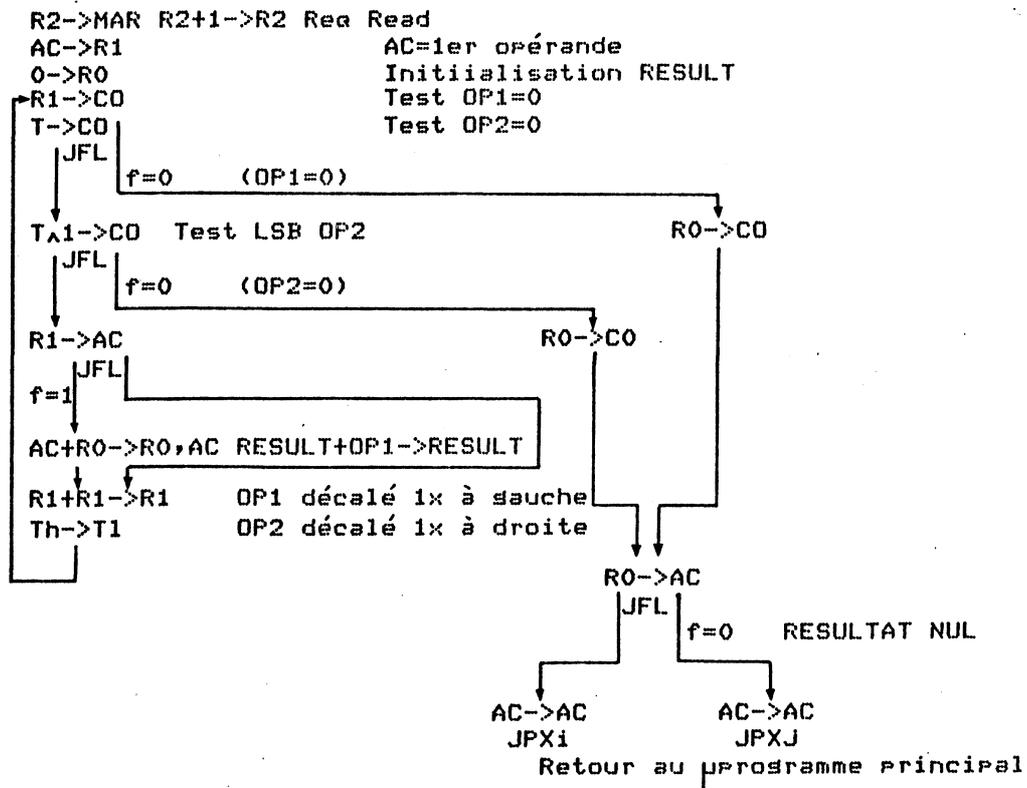
SUIT1 COPY,B,D
      ANDIM,B 1    Masque bit LSB
      JZRO,B SUIT2 Test bit LSB (=0->SUIT2)
      ADD,A RESULT Résult + A
      ST,A RESULT  ->Résult

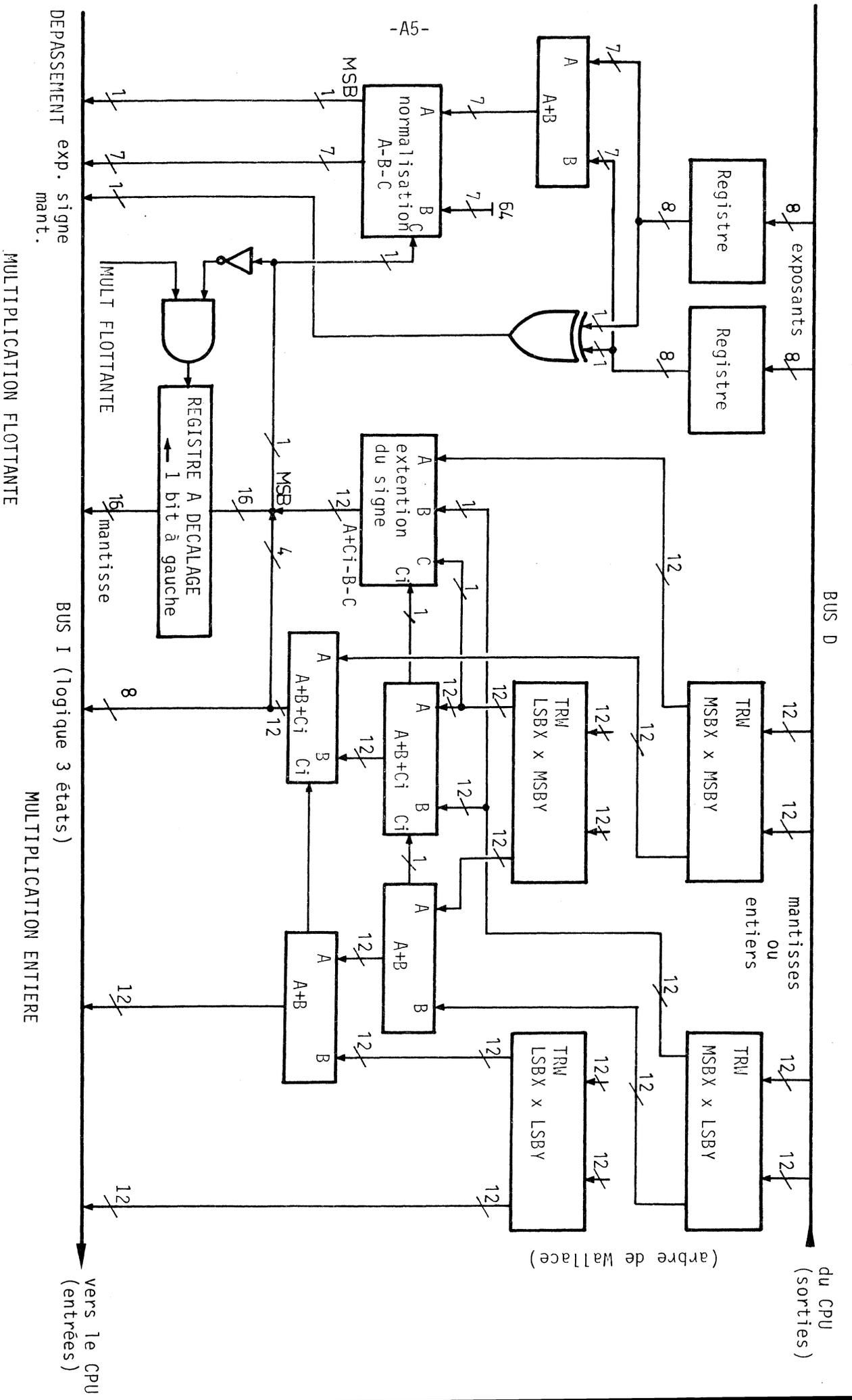
SUIT2 COPY,D,B
      SHR,B 1      B décalé 1 fois à droite
      JZRO,B RETO  Test B=0 ->fin de la multiplication
      SHL,A 1      A décalé 1 fois à gauche
      JMP SUIT1    Retour à SUIT1

RETO LD,A RESULT  Résult -> A
RET1 RETURN       Fin du sous-programme : retour au
                  programme principal
```

Sous/µprogramme MULT





BLOC DIAGRAMME DE LA MULTIPLICATION CABLEE ENTIERE ET FLOTTANTE

ANNEXE B

Fonctions CPE
Codes de Saut MCU

F-GROUP	R-GROUP	MICRO-FUNCTION
0	I II III	$R_n + (AC \wedge K) + CI \rightarrow R_n \vee AC$ $M + (AC \wedge K) + CI \rightarrow AT$ $AT_L \wedge (I_L \wedge K_L) \rightarrow RO$ $LIV \vee (I_H \wedge K_H) \wedge AT_H J \rightarrow AT_H$ $EAT_L \wedge (I_L \wedge K_L) \vee EAT_H \wedge (I_H \wedge K_H) J \rightarrow AT_H$
1	I II III	$K \vee R_n \rightarrow MAR$ $R_n + CI + K \rightarrow R_n$ $K \vee M \rightarrow MAR$ $M + CI + K \rightarrow AT$ $(AT \vee K) + (AT \wedge K) + CI \rightarrow AT$
2	I II III	$(AC \wedge K) - 1 + CI \rightarrow R_n$ $(AC \wedge K) - 1 + CI \rightarrow AT$ $(I \wedge K) - 1 + CI \rightarrow AT$
3	I II III	$R_n + (AC \wedge K) + CI \rightarrow R_n$ $M + (AC \wedge K) + CI \rightarrow AT$ $AT + (I \wedge K) + CI \rightarrow AT$
4	I II III	$CI \vee (R_n \wedge AC \wedge K) \rightarrow CO$ $R_n \wedge (AC \wedge K) \rightarrow R_n$ $CI \vee (M \wedge AC \wedge K) \rightarrow CO$ $M \wedge (AC \wedge K) \rightarrow AT$ $CI \vee (AT \wedge I \wedge K) \rightarrow CO$ $AT \wedge (I \wedge K) \rightarrow AT$
5	I II III	$CI \vee (R_n \wedge K) \rightarrow CO$ $K \wedge R_n \rightarrow R_n$ $CI \vee (M \wedge K) \rightarrow CO$ $K \wedge M \rightarrow AT$ $CI \vee (AT \wedge K) \rightarrow CO$ $K \wedge AT \rightarrow AT$
6	I II III	$CI \vee (AC \wedge K) \rightarrow CO$ $R_n \vee (AC \wedge K) \rightarrow R_n$ $CI \vee (AC \wedge K) \rightarrow CO$ $M \vee (AC \wedge K) \rightarrow AT$ $CI \vee (I \wedge K) \rightarrow CO$ $AT \vee (I \wedge K) \rightarrow AT$
7	I II III	$CI \vee (R_n \wedge AC \wedge K) \rightarrow CO$ $R_n \oplus (AC \wedge K) \rightarrow R_n$ $CI \vee (M \wedge AC \wedge K) \rightarrow CO$ $M \oplus (AC \wedge K) \rightarrow AT$ $CI \vee (AT \wedge I \wedge K) \rightarrow CO$ $AT \oplus (I \wedge K) \rightarrow AT$

SYMBOL	MEANING
I, K, M	Data on the I, K, M busses
R _n	Contents of register n (R-Group 1)
AC	Contents of the accumulator
AT	Contents of AC or T, as specified
CI	Data on the carry input
CO	Data on the carry output
L, H	As subscripts, designate low and high order bit
+	2's complement addition
-	2's complement subtraction
∧	Logical AND
∨	Logical OR
⊕	Exclusive NOR
->	Deposit into

MNEMONIC DESCRIPTION	AC FUNCTION							NEXT ROW					NEXT COLUMN			
	6	5	4	3	2	1	0	8	7	6	5	4	3	2	1	0
JCC Jump in Current Column	0	0	d ₄	d ₃	d ₂	d ₁	d ₀	d ₄	d ₃	d ₂	d ₁	d ₀	m ₃	m ₂	m ₁	m ₀
JZR Jump in Zero Row	0	1	0	d ₃	d ₂	d ₁	d ₀	0	0	0	0	0	d ₃	d ₂	d ₁	d ₀
JCR Jump in Current Row	0	1	1	d ₃	d ₂	d ₁	d ₀	m ₈	m ₇	m ₆	m ₅	m ₄	d ₃	d ₂	d ₁	d ₀
JCE Jump in Column/Enable	1	1	1	0	d ₂	d ₁	d ₀	m ₈	m ₇	d ₂	d ₁	d ₀	m ₃	m ₂	m ₁	m ₀
JFL Jump/test F-latch	1	0	0	d ₃	d ₂	d ₁	d ₀	m ₈	d ₃	d ₂	d ₁	d ₀	m ₃	0	1	f
JCF Jump/test C-flag	1	0	1	0	d ₂	d ₁	d ₀	m ₈	m ₇	d ₂	d ₁	d ₀	m ₃	0	1	c
JZF Jump/test Z-flag	1	0	1	1	d ₂	d ₁	d ₀	m ₈	m ₇	d ₂	d ₁	d ₀	m ₃	0	1	z
JPR Jump/test PR-latches	1	1	0	0	d ₂	d ₁	d ₀	m ₈	m ₇	d ₂	d ₁	d ₀	p ₃	p ₂	p ₁	p ₀
JLL Jump/test Left PR-bits	1	1	0	1	d ₂	d ₁	d ₀	m ₈	m ₇	d ₂	d ₁	d ₀	0	1	p ₃	p ₂
JRL Jump/test Right PR-bits	1	1	1	1	1	d ₁	d ₀	m ₈	m ₇	1	d ₁	d ₀	1	1	p ₁	p ₀
JPX Jump/test PX-bus	1	1	1	1	1	d ₁	d ₀	m ₈	m ₇	m ₆	d ₁	d ₀	x ₇	x ₆	x ₅	x ₄

d_n Data on adress control line n
m_n Data in microprogram adress register bit n
p_n Data in PR-latch bit n
x_n Data on PX-bus line n
f,c,z Contents of F-latch,C-flag,Z-flag

Code opération

x ₇	x ₆	x ₅	x ₄	p ₃	p ₂	p ₁	p ₀
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

CODES DE SAUT MCU

ANNEXE C

Jeu d'instructions GESPRO

JEU D'INSTRUCTIONS GESPRO version 82

- Rn : 4 registres généraux A (R0) , B (R1) , D (R2) , E (R3)
- RJ : 2 registres accumulateurs A (R0) , B (R1)
- Ri : 2 registres de base pour l'indexation ID (R2) , IE (R3)
- @ : signe d'indirection (indique que l'instruction existe en adressage direct et indirect)
- CONF , MASK : donnée sur 24 bits
- DATAL : donnée sur 16 bits
- DATA H,L: donnée sur 8+16 bits
- NBRE : nombre compris entre 0 et 24 , indiquant le nombre de décalages à effectuer dans les instructions de rotation et de décalage
- SADR : adresse SOURCE
- DADR : adresse DESTINATION
- SRn : registre SOURCE
- DRn : registre DESTINATION dans les instructions de transfert
- FRn , FA , F(ADR) : donnée FLOTTANTE
- EA , EB : donnée ENTIERE

**** CODE OPERATION en OCTAL ***

MNEMONIQUE	CODE OPERATION	FONCTION	FORMAT					
<u>MEMORISATION</u>								
ST,Rn	@JADR	104 + Rn	Rn -> [(ADR)]	<table border="1"> <tr> <td>ST,Rn</td> <td>@</td> <td>ADR</td> </tr> </table>	ST,Rn	@	ADR	
ST,Rn	@	ADR						
STI,RJ,Ri	ADR	302 + RJ	RJ -> (Ri + ADR)	<table border="1"> <tr> <td>STI,RJ</td> <td>Ri</td> <td>ADR</td> </tr> </table>	STI,RJ	Ri	ADR	
STI,RJ	Ri	ADR						
STIM	ADR,DATA	253	DATA -> (ADR)	<table border="1"> <tr> <td>STIM</td> <td>ADR</td> </tr> <tr> <td colspan="2">DATA</td> </tr> </table>	STIM	ADR	DATA	
STIM	ADR							
DATA								
STM,Rn	ADR,DATA	264 + Rn	DATA -> (ADR) -> (ADR + 1) . . -> (ADR+(Rn-1))	<table border="1"> <tr> <td>STM</td> <td>ADR</td> </tr> <tr> <td colspan="2">DATA</td> </tr> </table>	STM	ADR	DATA	
STM	ADR							
DATA								
<u>CHARGEMENT</u>								
LD,Rn	@JADR	320 + Rn	[(ADR)] -> Rn	<table border="1"> <tr> <td>LD,Rn</td> <td>@</td> <td>ADR</td> </tr> </table>	LD,Rn	@	ADR	
LD,Rn	@	ADR						
LDI,RJ,Ri	ADR	300 + Rn	(Ri+ADR) -> RJ	<table border="1"> <tr> <td>LDI,RJ</td> <td>Ri</td> <td>ADR</td> </tr> </table>	LDI,RJ	Ri	ADR	
LDI,RJ	Ri	ADR						
LDD,A	@JADR	230	[(ADR)] -> A [(ADR)+1] -> B	<table border="1"> <tr> <td>LDD,A</td> <td>@</td> <td>ADR</td> </tr> </table>	LDD,A	@	ADR	
LDD,A	@	ADR						
LDD,IE	@JADR	231	[(ADR)] -> D [(ADR)+1] -> E	<table border="1"> <tr> <td>LDD,IE</td> <td>@</td> <td>ADR</td> </tr> </table>	LDD,IE	@	ADR	
LDD,IE	@	ADR						
LIM,Rn	DATA	200 + Rn	DATA -> Rn	<table border="1"> <tr> <td>LIM,Rn</td> <td>0</td> </tr> <tr> <td colspan="2">DATA</td> </tr> </table>	LIM,Rn	0	DATA	
LIM,Rn	0							
DATA								

IML,Rn	DATA H,L		DATA H,L -> Rn	<table border="1"> <tr><td>IML,Rn</td><td>0</td></tr> <tr><td>DATAL</td><td>DATAL</td></tr> </table>	IML,Rn	0	DATAL	DATAL		
IML,Rn	0									
DATAL	DATAL									
SETA,Rn	DATAL	240 + Rn	DATAL -> Rn	<table border="1"> <tr><td>SETA,Rn</td><td>DATAL</td></tr> </table>	SETA,Rn	DATAL				
SETA,Rn	DATAL									
MOVE	SADR,DADR,NBMOT	44	(SADR+i)->(DADR+i) i=0,NBMOT-1	<table border="1"> <tr><td>MOVE</td><td>SADR</td></tr> <tr><td></td><td>DADR</td></tr> <tr><td></td><td>NBMOT</td></tr> </table>	MOVE	SADR		DADR		NBMOT
MOVE	SADR									
	DADR									
	NBMOT									

transfert de tableau a tableau disjoints ou non disjoints

OPERATIONS ARITHMETIQUES

Addition

ADD,Rn	[@]ADR	334 + Rn	Rn+[ADR] -> Rn	<table border="1"> <tr><td>ADD,Rn</td><td>@ ADR</td></tr> </table>	ADD,Rn	@ ADR		
ADD,Rn	@ ADR							
ADDI,RJ,Ri	ADR	314 + RJ	RJ+(Ri+ADR) -> RJ	<table border="1"> <tr><td>ADDI,RJ</td><td>Ri ADR</td></tr> </table>	ADDI,RJ	Ri ADR		
ADDI,RJ	Ri ADR							
ADIM,RJ	DATA	212 + RJ	RJ+DATA -> RJ	<table border="1"> <tr><td>ADIM,RJ</td><td>0</td></tr> <tr><td></td><td>DATA</td></tr> </table>	ADIM,RJ	0		DATA
ADIM,RJ	0							
	DATA							
IMADD,RJ	DATA H,L		RJ+DATA H,L -> RJ	<table border="1"> <tr><td>IMADD,RJ</td><td>0</td></tr> <tr><td>DATAL</td><td>DATAL</td></tr> </table>	IMADD,RJ	0	DATAL	DATAL
IMADD,RJ	0							
DATAL	DATAL							
RADD,RJ		300 + RJ	si J=A A+B -> A si J=B B+A -> B	<table border="1"> <tr><td>RADD,RJ</td><td>0</td></tr> </table>	RADD,RJ	0		
RADD,RJ	0							
COPY,SRn,DRn	DATAL	340 + SRn*4 + DRn	SRn+DATAL->SRn,DRn	<table border="1"> <tr><td>COPY,SRn,DRn</td><td>DATAL</td></tr> </table>	COPY,SRn,DRn	DATAL		
COPY,SRn,DRn	DATAL							

Soustraction

SUB,Rn	[@]ADR	324 + Rn	Rn-[ADR] -> Rn	<table border="1"> <tr><td>SUB,Rn</td><td>@ ADR</td></tr> </table>	SUB,Rn	@ ADR		
SUB,Rn	@ ADR							
SUBI,RJ,Ri	ADR	316 + RJ	RJ-(Ri+ADR) -> RJ	<table border="1"> <tr><td>SUBI,RJ</td><td>Ri ADR</td></tr> </table>	SUBI,RJ	Ri ADR		
SUBI,RJ	Ri ADR							
SUBIM,RJ	DATA	207 + 8*RJ	RJ-DATA -> RJ	<table border="1"> <tr><td>SUBIM,RJ</td><td>0</td></tr> <tr><td></td><td>DATA</td></tr> </table>	SUBIM,RJ	0		DATA
SUBIM,RJ	0							
	DATA							
IMSUB,RJ	DATA H,L		RJ-DATA H,L -> RJ	<table border="1"> <tr><td>IMSUB,RJ</td><td>0</td></tr> <tr><td>DATAL</td><td>DATAL</td></tr> </table>	IMSUB,RJ	0	DATAL	DATAL
IMSUB,RJ	0							
DATAL	DATAL							
RSUB,RJ		362 + RJ	si J=A A-B -> A si J=B B-A -> B	<table border="1"> <tr><td>RSUB,RJ</td><td>0</td></tr> </table>	RSUB,RJ	0		
RSUB,RJ	0							

Complémentation

CP,Rn		364 + Rn	\overline{Rn} -> Rn	<table border="1"> <tr><td>CP,Rn</td><td>0</td></tr> </table>	CP,Rn	0
CP,Rn	0					

OPERATIONS LOGIQUES

ET

AND,RJ	[@]ADR	220 + RJ	RJ^[ADR] -> RJ	<table border="1"> <tr><td>AND,RJ</td><td>@ ADR</td></tr> </table>	AND,RJ	@ ADR		
AND,RJ	@ ADR							
ANDI,RJ,Ri	ADR	306 + RJ	RJ^(Ri+ADR) -> RJ	<table border="1"> <tr><td>ANDI,RJ</td><td>Ri ADR</td></tr> </table>	ANDI,RJ	Ri ADR		
ANDI,RJ	Ri ADR							
ANDIM,RJ	DATA	204 + 8*RJ	RJ^DATA -> RJ	<table border="1"> <tr><td>ANDIM,RJ</td><td>0</td></tr> <tr><td></td><td>DATA</td></tr> </table>	ANDIM,RJ	0		DATA
ANDIM,RJ	0							
	DATA							
IMAND,RJ	DATA H,L		RJ^DATA H,L -> RJ	<table border="1"> <tr><td>IMAND,RJ</td><td>0</td></tr> <tr><td>DATAL</td><td>DATAL</td></tr> </table>	IMAND,RJ	0	DATAL	DATAL
IMAND,RJ	0							
DATAL	DATAL							
RAND,Rn		120 + 4*Rn	A^Rn -> A	<table border="1"> <tr><td>RAND,Rn</td><td>0</td></tr> </table>	RAND,Rn	0		
RAND,Rn	0							

OU

OR,RJ	[@]ADR	222 + RJ	RJv[(ADR)] -> RJ	OR,RJ	@ ADR
ORI,RJ,Ri	ADR	310 + RJ	RJv(Ri+ADR) -> RJ	ORI,RJ	Ri ADR
ORIM,RJ	DATA	205 + 8*RJ	RJvDATA -> RJ	ORIM,RJ	0
				DATA	
IMOR,RJ	DATA H,L	.	RJvDATA H,L-> RJ	IMOR,RJ	0
				DATAH	DATAL
ROR,Rn		121 + 4*Rn	AvRn -> A	ROR,Rn	0

OU EXCLUSIF

XOR,RJ	[@]ADR	224 + RJ	RJ@(ADR)] -> RJ	XOR,RJ	@ ADR
XORI,RJ,Ri	ADR	312 + RJ	RJ@(Ri+ADR) -> RJ	XORI,RJ	Ri ADR
XORIM,RJ	DATA	206 + 8*RJ	RJ@DATA -> RJ	XORIM,RJ	0
				DATA	
IMXOR,RJ	DATA H,L	.	RJ@DATA H,L-> RJ	IMXOR,RJ	0
				DATAH	DATAL
RXOR,Rn		122 + 4*Rn	A@Rn -> A	RXOR,Rn	0

INCREMENTATION

INM	[@]ADR	62	[(ADR)]+1 -> [(ADR)] RAC+1 si [(ADR)]=0	INM	@ ADR
				[(ADR)]=0	
INMI,Ri	ADR	304	(Ri+ADR)+1 -> (Ri+ADR) RAC+1 si (Ri+ADR)=0	INMI	Ri ADR
				(Ri+ADR)=0	
IRJ,Rn	ADR	21 + 4*Rn	Rn+1 -> Rn saut à ADR si Rn=0	IRJ,Rn	ADR

DECREMENTATION

DCM	[@]ADR	60	[(ADR)]-1 -> [(ADR)] RAC+1 si [(ADR)]=0	DCM	@ ADR
				[(ADR)]=0	
DCMI,Ri	ADR	305	(Ri+ADR)-1 -> (Ri+ADR) RAC+1 si (Ri+ADR)=0	DCMI	Ri ADR
				(Ri+ADR)=0	
DRJ,Rn	ADR	22 + 4*Rn	Rn-1 -> Rn saut à ADR si Rn=0	DRJ,Rn	ADR

TRAITEMENT DU BIT

CPZRO	[@]ADR,CONF	77	comparaison "0"[(ADR)] et "0"CONF RAC+2 si "0"[(ADR)]# "0"(CONF)	CPZRO	@ ADR
				CONF	
				"0"[(ADR)]# "0"(CONF)	
CPONE	[@]ADR,CONF	73	comparaison "1"[(ADR)] et "1"CONF RAC+2 si "1"[(ADR)]# "1"(CONF)	CPONE	@ ADR
				CONF	
				"1"[(ADR)]# "1"(CONF)	
SETZRO	[@]ADR,CONF	70	[(ADR)],CONF->[(ADR)]	SETZRO	@ ADR
				CONF	

SETONE	[@]ADR,CONF	74	[(ADR)] v CONF -> [(ADR)]	<table border="1"> <tr> <td>SETONE</td> <td>@ : ADR</td> </tr> <tr> <td colspan="2">CONF</td> </tr> </table>	SETONE	@ : ADR	CONF	
SETONE	@ : ADR							
CONF								

TRAITEMENT DU MOT

Décalage à gauche

SHLA	NBRE	373	A0->A1, ..., A22->A23 0 -> A0	<table border="1"> <tr> <td>SHLA</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	SHLA	0	NBRE	
SHLA	0							
NBRE								

SHLB	NBRE	370	B0->B1, ..., B22->B23 0 -> B0	<table border="1"> <tr> <td>SHLB</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	SHLB	0	NBRE	
SHLB	0							
NBRE								

Décalage à droite

SHRA	NBRE	371	A23->A22, ..., A1->A0 0 -> A23	<table border="1"> <tr> <td>SHRA</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	SHRA	0	NBRE	
SHRA	0							
NBRE								

SHRB	NBRE	372	B23->B22, ..., B1->B0 0 -> B23	<table border="1"> <tr> <td>SHRB</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	SHRB	0	NBRE	
SHRB	0							
NBRE								

Rotation à gauche

ROLA	NBRE	377	A0->A1, ..., A22->A23 A23->A0	<table border="1"> <tr> <td>ROLA</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	ROLA	0	NBRE	
ROLA	0							
NBRE								

ROLB	NBRE	376	B0->B1, ..., B22->B23 B23->B0	<table border="1"> <tr> <td>ROLB</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	ROLB	0	NBRE	
ROLB	0							
NBRE								

Rotation à droite

RORA	NBRE	375	A23->A22, ..., A1->A0 A0->A23	<table border="1"> <tr> <td>RORA</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	RORA	0	NBRE	
RORA	0							
NBRE								

RORB	NBRE	374	B23->B22, ..., B1->B0 B0->B23	<table border="1"> <tr> <td>RORB</td> <td>0</td> </tr> <tr> <td colspan="2">NBRE</td> </tr> </table>	RORB	0	NBRE	
RORB	0							
NBRE								

LBYT, Rn	[@]ADR, MASK	330 + Rn	[(ADR)] o rotation à droite de 8 positions ^ MASK -> Rn	<table border="1"> <tr> <td>LBYT, Rn</td> <td>@ : ADR</td> </tr> <tr> <td colspan="2">MASK</td> </tr> </table>	LBYT, Rn	@ : ADR	MASK	
LBYT, Rn	@ : ADR							
MASK								

INSTRUCTIONS DE SAUT

JMP	[@]ADR	42	saut à [(ADR)]	<table border="1"> <tr> <td>JMP</td> <td>@ : ADR</td> </tr> </table>	JMP	@ : ADR
JMP	@ : ADR					

JMPI, Rn	[@]ADR	2 + 4*Rn	saut à [(ADR) + Rn]	<table border="1"> <tr> <td>JMPI, Rn</td> <td>@ : ADR</td> </tr> </table>	JMPI, Rn	@ : ADR
JMPI, Rn	@ : ADR					

JZRO, Rn	[@]ADR	20 + 4*Rn	saut à [(ADR)] si Rn=0	<table border="1"> <tr> <td>JZRO, Rn</td> <td>@ : ADR</td> </tr> </table>	JZRO, Rn	@ : ADR
JZRO, Rn	@ : ADR					

JNZR, Rn	[@]ADR	23 + 4*Rn	saut à [(ADR)] si Rn < 0	<table border="1"> <tr> <td>JNZR, Rn</td> <td>@ : ADR</td> </tr> </table>	JNZR, Rn	@ : ADR
JNZR, Rn	@ : ADR					

JTC	[@]ADR	50	saut à [(ADR)] si CARRY=0	<table border="1"> <tr> <td>JTC</td> <td>@ : ADR</td> </tr> </table>	JTC	@ : ADR
JTC	@ : ADR					

JMPM (CALL)	[@]ADR	63	saut à [(ADR)] + 1 avec marquage: RAC+1 -> [(ADR)]	<table border="1"> <tr> <td>JMPM</td> <td>@ : ADR</td> </tr> </table>	JMPM	@ : ADR
JMPM	@ : ADR					

TRAITEMENT DES INTERRUPTIONS

ION		140	validation des interruptions	ION	0
IOFF		141	invalidation des interruptions	IOFF	0
MSKAL		142	masquage de tous les niveaux	MSKAL	0
CLAL		143	désactivation de tous les niveaux	CLAL	0
MSKLV	NIV	144	masquage du niveau NIV	MSKLV	NIV
DSKLV	NIV	145	démasquage du niveau NIV	DSKLV	NIV
CCLV	NIV	146	désactivation du niveau NIV	CCLV	NIV
ACTLV	NIV	147	activation du niveau NIV	ACTLV	NIV
RCLV	NIV	166	lecture du niveau courant (NIV COUR) -> A	RCLV	4000
GIVUP		167	quitte le niveau courant en le désactivant	GIVUP	4000

INSTRUCTIONS MULTIPLIEUR

FLOAD,Rn	ADR,OVF	260 + Rn	Addition flottante: FRn + F(ADR) -> FRn RAC+1 si OVERFLOW	FLOAD,Rn	ADR
				JMP	OVF
FLOADM,Rn	ADR,OVF	"	Addition flottante: FRn + F(ADR) -> FRn RAC+1 si OVERFLOW	FLOADM,Rn	ADR
				JMPM	OVF
FLMUL,RJ	ADR,OVF	270 + RJ	Multipliation flottante: FRJ * F(ADR) -> FRJ RAC+1 si OVERFLOW	FLMUL,RJ	ADR
				JMP	OVF
FLMULM,RJ	ADR,OVF	"	Multipliation flottante: FRJ * F(ADR) -> FRJ RAC+1 si OVERFLOW	FLMULM,RJ	ADR
				JMPM	OVF
INTMUL		275	Multipliation entiere: EA * EB -> EA,EB	INTMUL	0
FLOINT	ADR	276	Conversion Flottant->Entier: FA -> EA saut a ADR si OVERFLOW	FLOINT	ADR
INTFLO		277	Conversion Entier->Flottant: EA -> FA	INTFLO	0

INSTRUCTIONS NA10

LDPT	LIMIT	41	Transfert des mots PT et test PT>=LIMIT	LDPT	LIMIT
MASS	LIMIT,NBSX<2	54	Calcul de masse	MASS	LIMIT
CPMASS	LIMIT,SUP,EQU	245	Comparaison Masse-Limit	JMP	NBSX<2
				CPMASS	0
				LIMIT	
				JMP	SUP
				JMP	EQU
STA	WAIT	55	Resynchronisation sur STA	STA	0
				JMP	WAIT
HOD01	EVBADR,ER67,ERAD,UNBIAS	57	Traitement Hodoscopes	HOD01	EVBADR
				JMP	ER67
				JMP	ERAD
				JMP	UNBIAS
WCT	LIMIT,INF,EQU	252	Test Word Count	WCT	LIMIT
				JMP	INF
				JMP	EQU
HOD02	EVBADR,ERAD,ERINC,RNTCO,RP1P2	53	Traitement Hodoscopes et TEST P1P2	HOD02	EVBADR
				JMP	ERAD
				JMP	ERINC
				JMP	RNTCO
				JMP	RP1P2
VR3R4	RVR3R4,RSIGN	246	Test VR3R4 et SIGN	VR3R4	176
				JMP	RVR3R4
				JMP	RSIGN
GOOD		250	Acquisition	GOOD	0
BAD		251	ReJet	BAD	0

PSEUDO-INSTRUCTIONS

ENTER entrée sous/programme
 RETURN fin sous/programme

DIRECTIVES

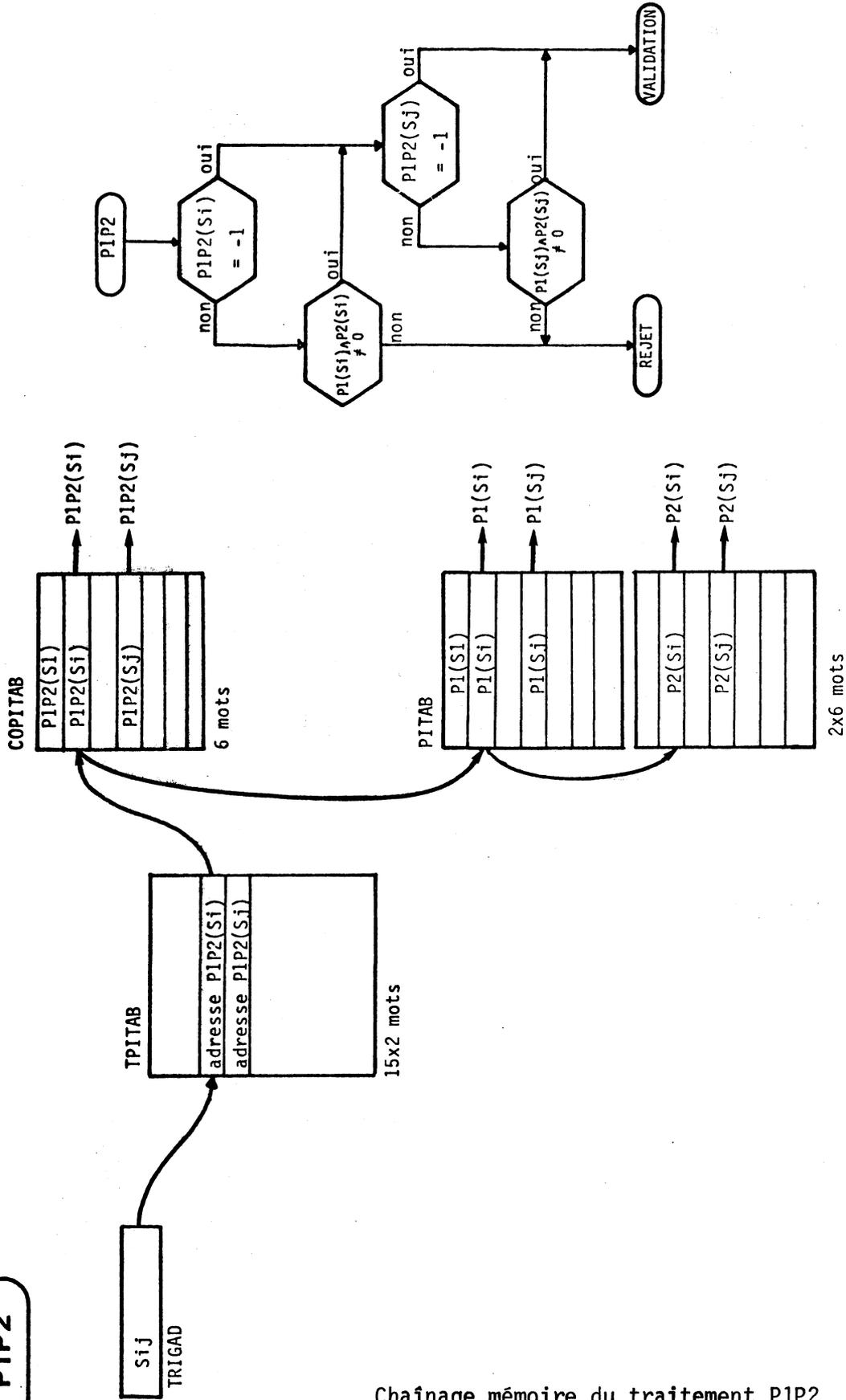
BEG définit le point d'entrée du programme
 EQU équivalence label-label ou label-valeur
 RES N réservation de N cases à partir de l'adresse courante
 END fin du programme (\$ NAME END)

ANNEXE D

ORGANIGRAMME DES TRAITEMENTS:

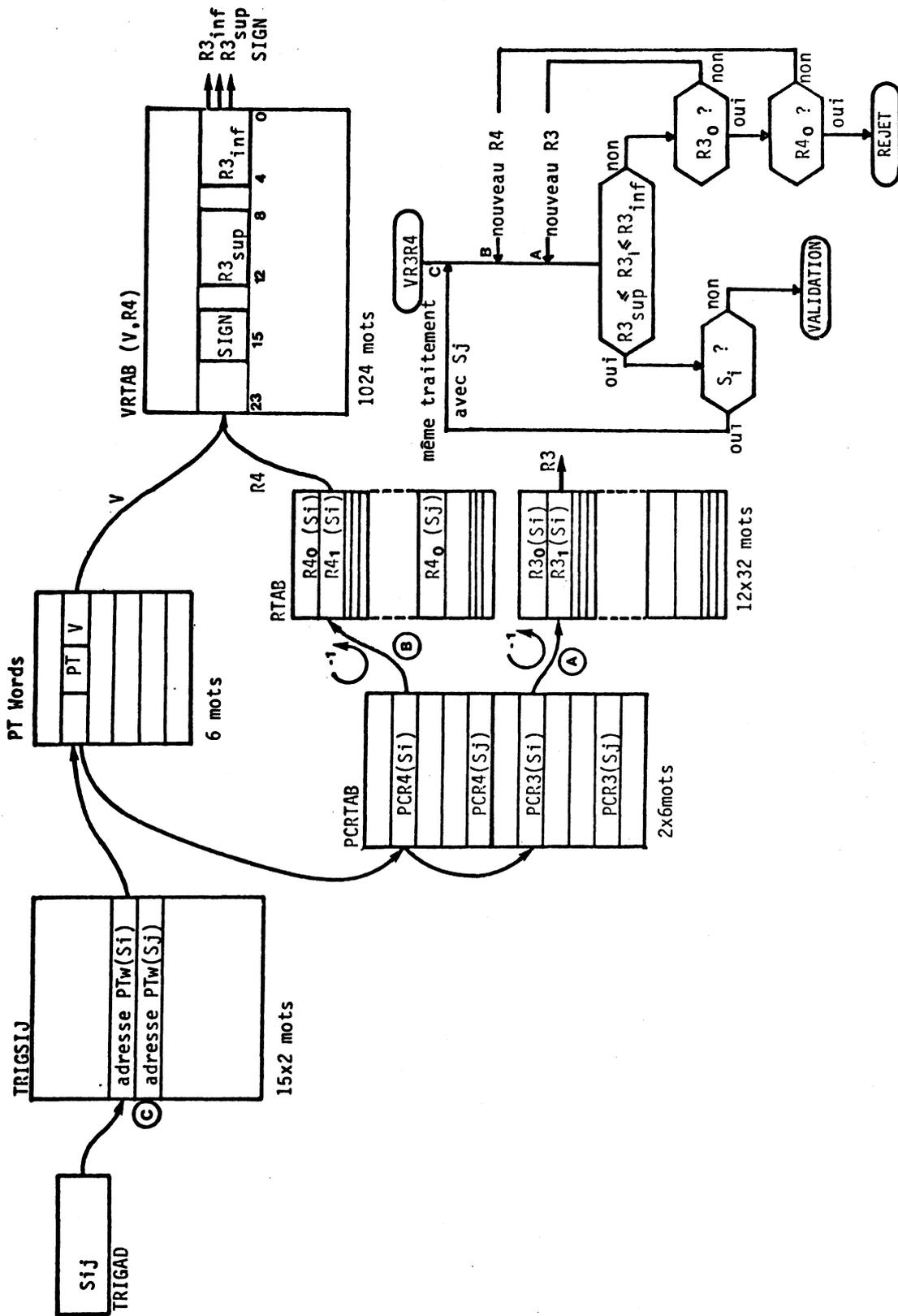
- P1P2
- VR3R4

P1P2



Chaînage mémoire du traitement P1P2

VR3R4



Chaînage mémoire du traitement VR3R4

REMERCIEMENTS

Ce travail a été effectué à la Division des Hautes Energies du Centre de Recherches Nucléaires de Strasbourg.

Je prie Monsieur Michel CROISSIAUX, Directeur de la Division, de bien vouloir trouver ici l'expression de ma gratitude pour l'accueil qu'il m'a réservé au sein de son équipe.

Je remercie Monsieur Gilbert WEILL de m'avoir fait l'honneur et l'amitié de présider le jury de thèse.

Je tiens à remercier Monsieur Gilbert SUTTER, d'avoir accepté de faire partie du jury.

Que Monsieur Jean-Marie MEYER, qui m'a fait bénéficier de son enseignement, m'a suggéré ce travail et l'a suivi en temps que Directeur de thèse, trouve ici l'expression de ma profonde reconnaissance.

Monsieur Jacques LECOQ m'a permis d'acquérir les connaissances nécessaires à ce travail et m'a prodigué ses conseils : je l'en remercie sincèrement.

Tout au long de ce travail, j'ai bénéficié de la confiance et de l'aide efficace et cordiale de Jean-Jacques BLAISING. Qu'il trouve ici le témoignage de ma très profonde reconnaissance.

Mes remerciements vont tout naturellement à Jean-Daniel BERST, toujours présent dans les moments difficiles. Son savoir-faire, ses conseils judicieux et son humour, ont permis de venir à bout de tous les écueils.

Jean Pierre FROBERGER a mené, avec moi, ce projet à terme. Je le remercie très sincèrement pour son aide efficace et amicale.

Je tiens également à remercier toutes les personnes qui de près ou de loin, m'ont aidé dans la réalisation de ce travail : physiciens, ingénieurs et techniciens, à Strasbourg comme au CERN.

Madame M.GOETZ a assuré la dactylographie de ce mémoire avec diligence et compétence : je l'en remercie.

REFERENCES

1. A HIGH RESOLUTION SPECTROMETER TO STUDY HIGH-MASS DIMUON PAIRS
NA10 Technical Paper CERN May 24, 1982
2. LE SPECTROMETRE A DIMUONS DE L'EXPERIENCE NA10, SON SYSTEME DE
DECLENCHEMENT
Thèse de 3^è cycle-A COC/1980 Université de PARIS VII
3. R M H Module Type 4236-Provisional Specifications
CERN-EP 9294-1A4/1981.08 05
4. CONTRIBUTION A L'ETUDE DES SYSTEMES INFORMATIQUES EN TEMPS REELS
EN PHYSIQUE DES PARTICULES ELEMENTAIRES
Thèse d'Etat-J.M MEYER/1977 Université du HAUT-RHIN
5. DATAWAY Specifications CERN-EP 199-11A4/1981.08.05

CAMAC-Organisation of multi-crates EURATOM Report EUR 4600E/1972
6. APPORT D'UN SYSTEME DE SEQUENCEURS MICROPROGRAMMES A L'ACQUISITION
ET AU FILTRAGE DE DONNEES EN PHYSIQUE DES PARTICULES
Thèse d'Etat-J LECOQ/1982 Université de HAUTE-ALSACE
7. USE OF FAST PROCESSORS FOR ON LINE TRIGGERING AND EVENT
FILTERING J D.BERST, J LECOQ, J.M.MEYER and M.PERRIN
(LEIN Mulhouse, LPNPP Strasbourg) Contribution to
TOPICAL CONFERENCE ON THE APPLICATION OF MICROPROCESSORS TO
HIGH-ENERGY PHYSICS EXPERIMENTS (CERN May 4-6, 1981)
CERN 81-07 July 17, 1981

MULTIPROCESSOR ACQUISITION SYSTEM INCLUDING A SPECIAL PURPOSE
CAMAC DEDICATED MICROPROCESSOR
J. LECOQ, J.M.MEYER, M.PERRIN, C.BOULIN, G.METZGER
Contribution to CAMAC SYMPOSIUM September 25-29, 1978 HAMBURG
CRN 78-16
8. FILTRAGE EN LIGNE DES EVENEMENTS SUR UNE EXPERIENCE DE PRODUCTION
DE DIMUONS DE HAUTE MASSE
Thèse de 3^è cycle-J.P.FROBERGER/1980 Université LOUIS PASTEUR
Strasbourg I

- 9 300 MICROPROGRAM CONTROL UNIT INTEL MCS 149-774-1K
- 3002 CONTROL PROCESS ELEMENT INTEL MCS 269-0275/27-5K
- 3214 PRIORITY INTERRUPT CONTROL UNIT INTEL MCS 396-0275/27-5K

10 ETUDE D'UN SYSTEME D'AIDE A LA MICROPROGRAMMATION
Thèse de 3^e cycle-M MARMONIER/1982 Université de HAUTE-ALSACE

11 CONCEPTION ET REALISATION D'UN SYSTEME DE DEVELOPPEMENT
EXECUTABLE SUR MINIORDINATEUR NORD-10 POUR UN MICROPROCESSEUR
SPECIALISE
Thèse de 3^e cycle-G HUOT/1978 Université LOUIS PASTEUR Strasbourg

Un système microprogrammable multiprocesseur de filtrage en ligne des dimuons de hautes masses a été développé comme trigger de second niveau dans le cadre de l'expérience NA10 au SPS du CERN. Cette thèse présente le matériel et décrit en détails le logiciel et les performances du filtrage.

A microprogrammable multiprocessor system, filtering on-line the high mass dimuons, has been developed as part of the second level trigger of the NA10 experiment at CERN-SPS. This thesis presents the hardware and describes in detail the software and the performances of the filtering.