

Ein Interface-System für die mehrparametrische Vielkanalanalyse bei Kernstrahlungsmessungen mit Siemens-Prozeßrechner R30

H.-J. Schuster, K. Weise

Physikalisch-Technische Bundesanstalt, Braunschweig

Nachdem in den vergangenen Jahren über die Hardware- und Softwareprodukte der Physikalisch-Technischen Bundesanstalt Braunschweig in allgemeiner Form vorgetragen wurde, soll heute die Automatisierung eines konkreten Prozesses mit Siemens-Rechnern vorgestellt werden. Es handelt sich um die Äquivalentdosis-messung in einem Neutronenfeld. An dieser Stelle wird über die Hardware- und in einem weiteren Vortrag ³⁾ über die Software-Implementierung dieses Prozesses gesprochen. Zunächst einiges über die physikalische Aufgabenstellung.

1 Meßaufgabe

Die durch ein Neutronenfeld im Körpergewebe erzeugte Äquivalentdosis hängt von der Energie und Richtung der Neutronen ab.

Zur Messung der Neutronenenergie wurde ein ³He-Spektrometer entwickelt. Es besteht aus zwei Oberflächensperrschichtdetektoren und einem zwischen ihnen angeordneten Proportionalzählrohr, das mit ³He-Gas gefüllt ist. Diese Detektor-Anordnung ist erforderlich, um die Neutronenenergien in einem weiten Energiebereich (0,1 - 12 MeV) zu messen und gleichzeitig die Störsignale der immer vorhandenen Gammastrahlung zu unterdrücken.

Zur Meßwerteerfassung müssen die Ausgangsimpulse der drei Detektoren digitalisiert und in den Rechner eingezählt werden (Inkrementmodus). Zusätzlich müssen sie, wenn eine Koinzidenzbedingung erfüllt ist, auch noch im Blockmodus in den Rechner übertragen werden.

Eine besondere Schwierigkeit ist dabei, daß diese Datenübertragungen mit hoher Rate erfolgen müssen und den Zentralprozessor wenig belasten dürfen. Die Datenübertragung wird mit dem PTB-Interface (Ein-/Ausgabe-Steuerung)²⁾ für Siemensrechner gelöst. Zunächst wird das Interface kurz beschrieben.

2 Standardfunktionen des PTB-Interface für Siemens-Prozeßrechner R30

Die in der PTB entwickelte Ein-/Ausgabe-(EA-)Steuerung (PTB-Interface) für Siemens-Prozeßrechner 300 ist modular aus Funktionsbausteinen aufgebaut. Sie befinden sich auf Steuerplatinen, die auf den Bus der EA-Steuerung gesteckt werden. Es gibt folgende Funktionsbausteine: Digitale Ein- und Ausgänge, Basisalarmeingänge, Zähler, Timer und einen sehr vielseitigen Analogbaustein.

Alle wichtigen Kommunikationssignale der Funktionsbausteine sind als Einzelkontakte am oberen Rand der Steuerplatinen verfügbar. Alle Einzelkontakte zusammen ergeben das interne Programmierfeld der EA-Steuerung, auf dem die Funktionsbausteine durch Steckleitungen beliebig verbunden und so die verschiedensten Hardwarekonfigurationen erzeugt werden können.

Die genannten Funktionsbausteine sind in ähnlicher Weise auch bei der Prozeßeinheit 3600¹⁾ implementiert bis auf den Basisalarmeingang, der das PTB-Interface für periphere Initiativen sehr viel flexibler macht. Im folgenden soll dieser Funktionsbaustein kurz beschrieben werden.

Über Basisalarmeingänge werden alle peripheren Initiativen (Organisationsalarme, Datenalarme) abgewickelt. Ein Basisalarmeingang besitzt die Kommunikationssignale PA (periphere Anforderung), \overline{QP} (Quittung an die Peripherie), \overline{NP} (peripheres Adressensignal oder Aufschaltsignal), F (Freigabesignal). Durch einen positiven Pegel am PA-Eingang wird eine periphere Anforderung beim Basisalarmeingang angemeldet. Das periphere Aufschaltsignal

\overline{NP} (Signalausgang) geht von 5V auf 0V, wenn die PA zum Rechner weitergeleitet wird und die Alarmbegleitinformation auf die Leitung zum Rechner geschaltet werden muß ²⁾. Mit \overline{NP} wird z.B. bei einer Dateneingabe der Dateneingang (DE) geöffnet und damit die einzugebende Information auf die DE-Leitungen geschaltet. Bei einem Organisationsalarm wird mit \overline{NP} der periphere Begleiter PBO ¹⁾ gesetzt und die Alarmnummer oder die Prozeßsignalförderadresse auf den DE-Bus geschaltet. Durch das Quittungssignal \overline{QP} (Signalausgang) wird die Bearbeitung einer PA abgeschlossen und die PA zurückgesetzt. Über das Freigabesignal F (Signaleingang) kann der Basisalarmeingang freigegeben oder gesperrt werden.

Im folgenden soll die Funktionsweise des PTB-Interface am Beispiel einer Dateneingabe im Blockmodus beschrieben werden. In Abb. 1 ist der Signal- oder Programmierplan für eine entsprechende Anordnung dargestellt. Für jede Interface-Konfiguration wird ein solcher Programmierplan als Arbeitsunterlage angefertigt.

Für eine Dateneingabe im Blockmodus werden im wesentlichen folgende Funktionsbausteine benötigt: Ein statischer Eingang, ein dynamischer Eingang für Organisationsalarme und zwei Basisalarmeingänge. Auf dem Signalplan sind die Kommunikationssignale dieser Funktionsbausteine eingezeichnet. Wenn der eine Basisalarmeingang, der auf der Grundplatine der EA-Steuerung vorgesehen ist, wie in diesem Fall nicht ausreicht, wird er mittels einer Alarmsynchronisationsplatine vervielfacht. Diese Platine belegt einen Basisalarmeingang und liefert acht neue Basisalarmeingänge. An den Basisalarmeingang 0 der Alarmsynchronisationsplatine wurde der dynamische Digitaleingang und an den Basisalarmeingang 1 der statische Digitaleingang 1 der Eingabeplatine angeschlossen.

Die Bedeutung der eingezeichneten Steckleitungen und Signale ergibt sich aus der Funktionsbeschreibung des dargestellten Interface-Signalplanes.

Die Dateneingabe wird durch die periphere Anforderung PA' eingeleitet. PA' wird über den Frontstecker und die Wahlleitung W1 an den Takteingang des Synchronisations-Flip-Flops des Digitaleinganges 1 geleitet. Jeder Digitaleingang besitzt ein solches Synchronisations-FF. Dieses Flip-Flop wird durch PA' gesetzt und mit seinem Ausgangssignal eine periphere Anforderung PA beim Basisalarmeingang 1 gestellt (Signalverbindung PA - PA1). Im Zuge der Bearbeitung dieser PA wird vom Aufschaltssignal $\overline{NP1}$ über das Adressierungssignal \overline{N} des Digitaleinganges die Information, die am Frontstecker des Einganges anliegt, auf den Bus zum Rechner geschaltet (Signalverbindung $\overline{NP1}$ - \overline{N}). Die Wertübernahme in den Rechner wird durch die Quittung $\overline{QP1}$ gemeldet. $\overline{QP1}$ setzt das Synchronisations-FF und damit die PA zurück (Signalverbindung $\overline{QP1}$ - \overline{QP}). Die Quittung wird auch durch das Signal QP' außen am Frontstecker angezeigt. Der Basisalarmeingang 1 ist nur solange freigegeben, wie das Signal PDAF (Freigabe für periphere Datenanforderungen) gesetzt ist (Signalverbindung PDAF - F1).

PDAF wird automatisch zurückgesetzt, wenn der Eingabepuffer im Arbeitsspeicher mit Werten voll geschrieben ist und der EA-Prozessor neu versorgt werden muß¹). Diese Tatsache muß dem Programm, das für die Versorgung des EA-Prozessors zuständig ist, durch einen Organisationsalarm mitgeteilt werden. Daher wird mit der Rückflanke des PDAF-Signals das Bit 0 (ALO) des dynamischen Einganges gesetzt (Signalverbindung PDAF - ALO). Der dynamische Eingang stellt dann automatisch eine POA (periphere Organisationsanforderung) beim Basisalarmeingang 0, an den der dynamische Eingang angeschlossen ist (Signalverbindung PA, \overline{QP} - PA0, \overline{QPO}). Die Bearbeitung der POA erfolgt im Prinzip wie die der PDA (Datenanforderung), wobei aber jetzt mit dem Aufschaltssignal NPO der periphere Begleiter PBO gesetzt wird zur Kennzeichnung der POA¹) und der Basisalarmeingang vom Signal POAF (POA-Freigabe) freigegeben wird (Signalverbindungen NPO - PBO, FO - POAF). Schließlich wird der dynamische Eingang auch durch eine zentrale Initiative abgefragt, um festzustellen, welches Bit gesetzt ist und die POA verursacht hat¹). Dazu wird dem dynamischen Eingang

die zentrale Adresse 0 zugeordnet (Signalverbindung $\overline{NO} - \overline{N}$).

Damit ist die peripher initiierte Datenübertragung über einen Eingang bereits vollständig beschrieben. Darüber hinaus ist aber auch schon der wesentliche Teil des Interface des betrachteten Spektrometerprozesses beschrieben, bei dem statt eines Dateneinganges mit peripherer Initiative sechs solche simultan arbeitende Dateneingänge mit peripherer Initiative einzurichten sind, was im folgenden begründet werden soll.

3 Realisierung der Interface-Funktionen für die mehrparametrische Vielkanalanalyse mit Hilfe des PTB-Interfaces

Entsprechend der Meßaufgabe müssen die Informationen von drei ADC im Inkrementmodus und zusätzlich, wenn eine Koinzidenzbedingung erfüllt ist, im Blockmodus in den Rechner peripher initiiert eingegeben werden. Dazu werden sechs Digitaleingänge mit nachgeschalteten Basisalarmeingängen für periphere Initiativen, so wie eben beschrieben, eingerichtet. An die Digitaleingänge 1, 2, 3 werden die ADC's 1, 2, 3 für die Dateneingabe im Inkrementmodus angeschaltet. An die Eingänge 4, 5, 6 werden die ADC's zusätzlich angeschlossen für die Dateneingabe im Blockmodus.

Zur Umschaltung zwischen Inkrement- und Blockeingabe ist eine kleine Spezialplatine erforderlich. Spezialplatinen, die z.B. einfache logische Verknüpfungen realisieren, sind ausdrücklich im System zugelassen. Sie werden auch über Einzelkontakte und Steckleitungen in das Interface-Programm integriert.

Die hier benötigte Spezialplatine ist im gestrichelten Rahmen der Abb. 2 dargestellt. Die Datenübertragungen im Inkrementmodus werden durch die peripheren Anforderungen PA1', PA2', PA3' der ADC's 1, 2, 3, die die Synchronisations-Flip-Flops der Digitaleingänge 1, 2, 3 setzen, eingeleitet. Die Bearbeitung erfolgt dann wie in Abschnitt 2 beschrieben mittels der Basisalarmein-

gänge 1, 2, 3 und wird durch die Quittungen QP1', QP2', QP3' an die ADC's abgeschlossen. Daß die peripheren Datenanforderungen jetzt im Gegensatz zu Abschnitt 2 im Inkrementmodus bedient werden, hat auf ihre Bearbeitung in der EA-Steuerung keinen Einfluß. Der Modus wird durch den EAP-Befehl der EA-Schnittstelle, an die die EA-Steuerung angeschlossen ist, festgelegt (s.u.)¹). Nach dem Erhalt seiner Quittung ist ein ADC bereit, einen neuen Impuls zu digitalisieren. Der alte Digitalwert geht verloren.

Im Falle einer Koinzidenz werden die Digitalwerte der ADC's durch Unterdrückung ihrer Quittungen QP1', QP2', QP3' festgehalten und zusätzlich im Blockmodus über die Digitaleingänge 4, 5, 6 eingegeben.

Elektronisch sieht das so aus: Durch ein Koinzidenzsignal wird das Koinzidenz-Flip-Flop gesetzt. Dadurch werden über die UND-Tore 1, 2, 3 die Quittungen QP1', QP2', QP3' an die ADC's unterdrückt. Die Quittungen der Basisalarmeingänge $\overline{QP1}$, $\overline{QP2}$, $\overline{QP3}$ werden in den Flip-Flops FF1, FF2, FF3, deren Ausgänge UND-verknüpft sind, gespeichert. Bei Erfüllung der UND-Bedingung sind die Digitalwerte der drei ADC's bereits im Inkrementmodus eingegeben und müssen nun noch im Blockmodus eingegeben werden. Dazu wird zunächst eine periphere Anforderung PA4' beim Digital-eingang 4 gestellt und daraufhin der Wert des ADC1 im Blockmodus eingegeben. Von den Quittungen $\overline{QP4}$ und $\overline{QP5}$ werden die Anforderungen PA5' und PA6' abgeleitet und dadurch die Digitalwerte der ADC's 2 und 3 im Blockmodus eingegeben. Die Quittung $\overline{QP6}$ setzt das Koinzidenz-Flip-Flop zurück, wodurch der Univibrator 123/1 getriggert wird und über die Tore 4, 5, 6 die Quittungen QP1', QP2', QP3' an die ADC's ausgegeben werden, die nun neue Impulse digitalisieren können.

Abschließend bedarf es noch eines Hinweises darüber, wieso die peripheren Anforderungen der Basisalarmeingänge 1, 2, 3 vom Rechner im Inkrementmodus und die der Basisalarmeingänge 4, 5, 6

im Blockmodus bearbeitet werden. Die EA-Steuerung ist an zwei EA-Schnittstellen des EA-Prozessors angeschlossen, wobei die erste mit dem EAP-Befehl "Einzählen" und die zweite mit dem Befehl "Blockeingabe" versorgt ist ²). Eine Weichenschaltung in der Rechneranschaltung leitet die peripheren Anforderungen der Basisalarmeingänge 1, 2, 3 zur ersten und die der Basisalarmeingänge 4, 5, 6 zur zweiten EA-Schnittstelle. Zur Unterscheidung dient der periphere Begleiter 1 (PB1), der bei den Anforderungen der Basisalarmeingänge 4, 5, 6 durch das Aufschalt-signal gesetzt (s. Abb. 2 und 3) und bei den Anforderungen der übrigen Basisalarmeingänge nicht gesetzt wird.

Der gesamte Programmierplan der EA-Steuerung für die mehrparametrige Vielkanalanalyse ist in Abb. 3 dargestellt. Die Bestückung der EA-Steuerung besteht aus einem dynamischen Eingang einer Alarmsynchronisationsplatine für acht Basisalarmeingänge, zwei Eingabeplatinen und der beschriebenen Spezialplatine.

Der ADC1 ist an die statischen Digitaleingänge 1, 4, der ADC2 an die Eingänge 2, 5 und der ADC3 an die Eingänge 3, 6 angeschlossen.

Die Digitaleingänge 1 bis 6 belegen die Basisalarmeingänge 1 bis 6 und der dynamische Digitaleingang den Basisalarmeingang 0 der Alarmsynchronisationsplatine. Auf die einzelnen Signalverbindungen, die bereits alle in den Abschnitten 2 und 3 funktionell begründet wurden, soll hier nicht weiter eingegangen werden. Die Abbildung des Programmierplanes soll hauptsächlich zeigen, daß ein komplexes Interface-Problem der Praxis mit relativ geringem Aufwand gelöst werden kann.

Zur Hardwarekonfiguration des Systems gehören auch noch einige Zähler und Timer sowie ein schneller grafischer Display für zweidimensionale Spektren. Auf diese Geräte soll nicht weiter eingegangen werden.

Literatur

- 1) Siemens System 300/16-Bit, Prozeßeinheit 3600 Prozeß-signalformer, Beschreibung Best.-Nr. ESTE 4 - 108 a/000.
- 2) Schuster, H.-J.: Funktionen und Analyse der Schaltung einer Prozeßdaten-Ein-/Ausgabe-Steuerung für Siemens-Prozeßrechner des Systems 300/16-Bit und Rechner mit IEC-Bus. Ber. PTB-FMRB-75, Physikalisch-Technische Bundesanstalt Braunschweig 1979.
- 3) Weise, K.; Kluge, H.; Rybarsch, R.; Schuster, H.-J.: Ein Programmsystem für die mehrparametrische Vielkanalanalyse bei Kernstrahlungsmessungen mit Siemens-Prozeßrechner R30. Ber. 12. Jahrestagung Anwenderkreis I Siemens-Prozeßrechner (herausgeg. v. K. Heim), Rechenzentrum Universität Karlsruhe 1981.

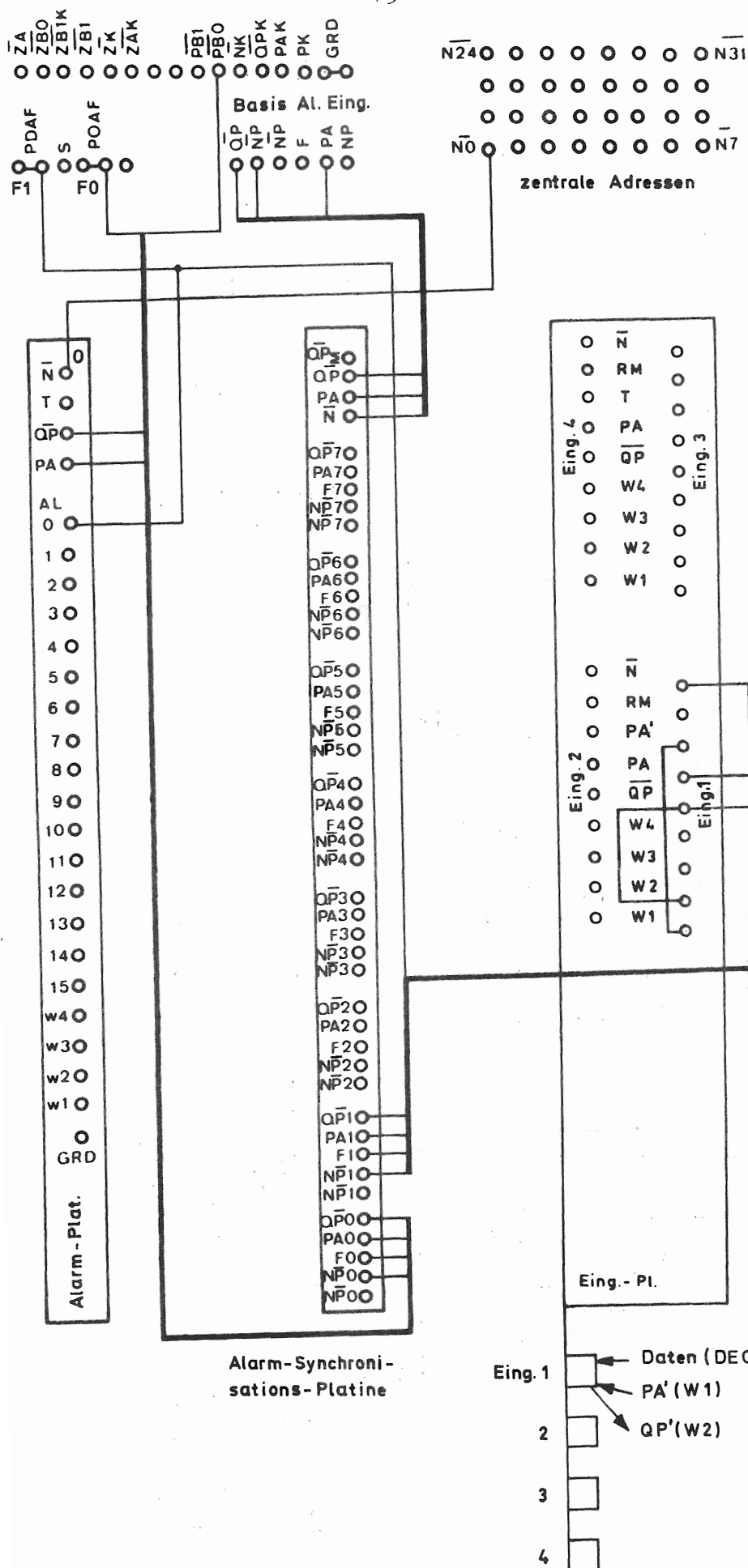


Abb.1: Signalplan für eine Dateneingabe im Blockmodus

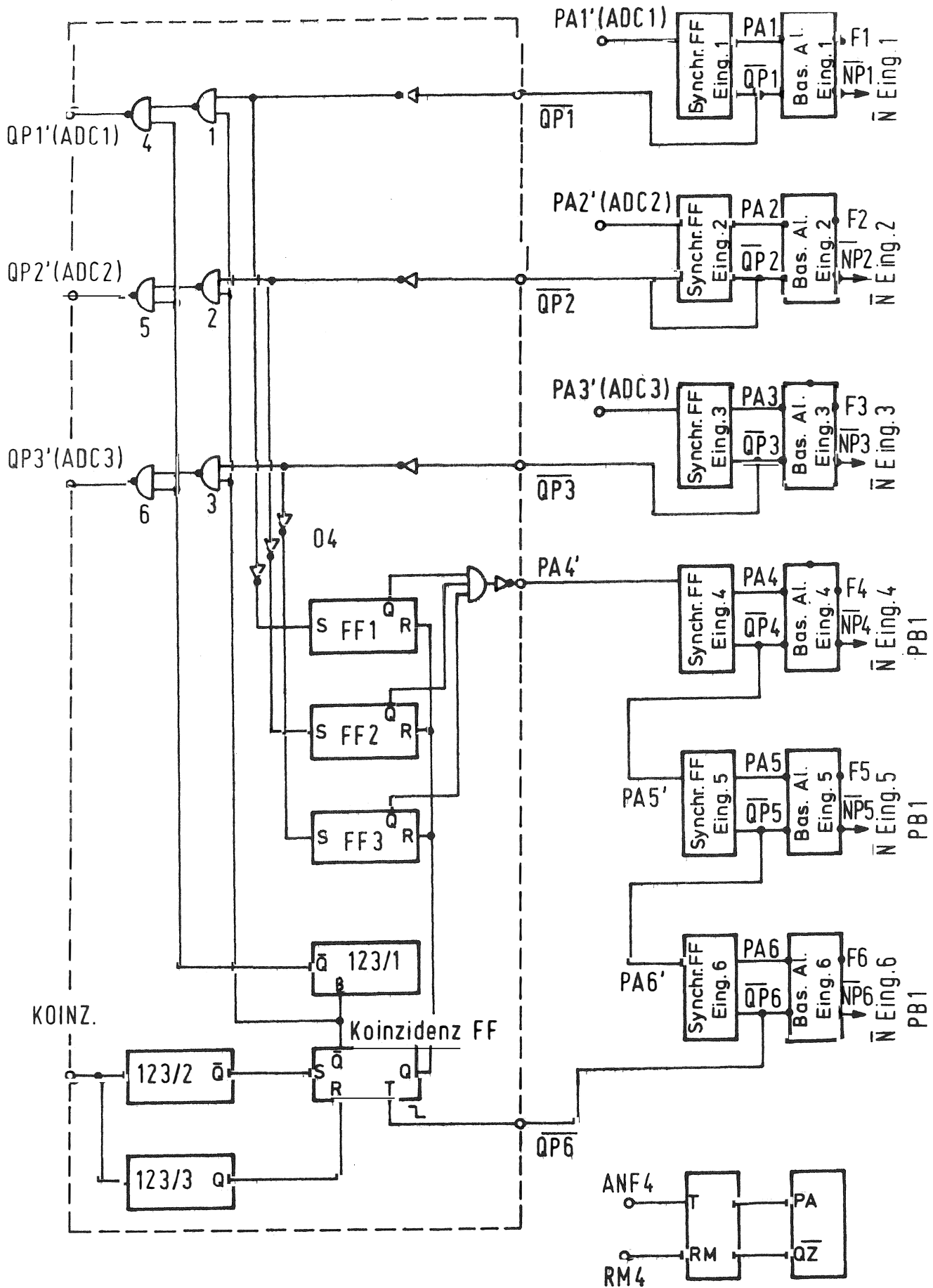


Abb. 2: Spezialplatine zur dreifachen Inkrement- und Blockeingabe

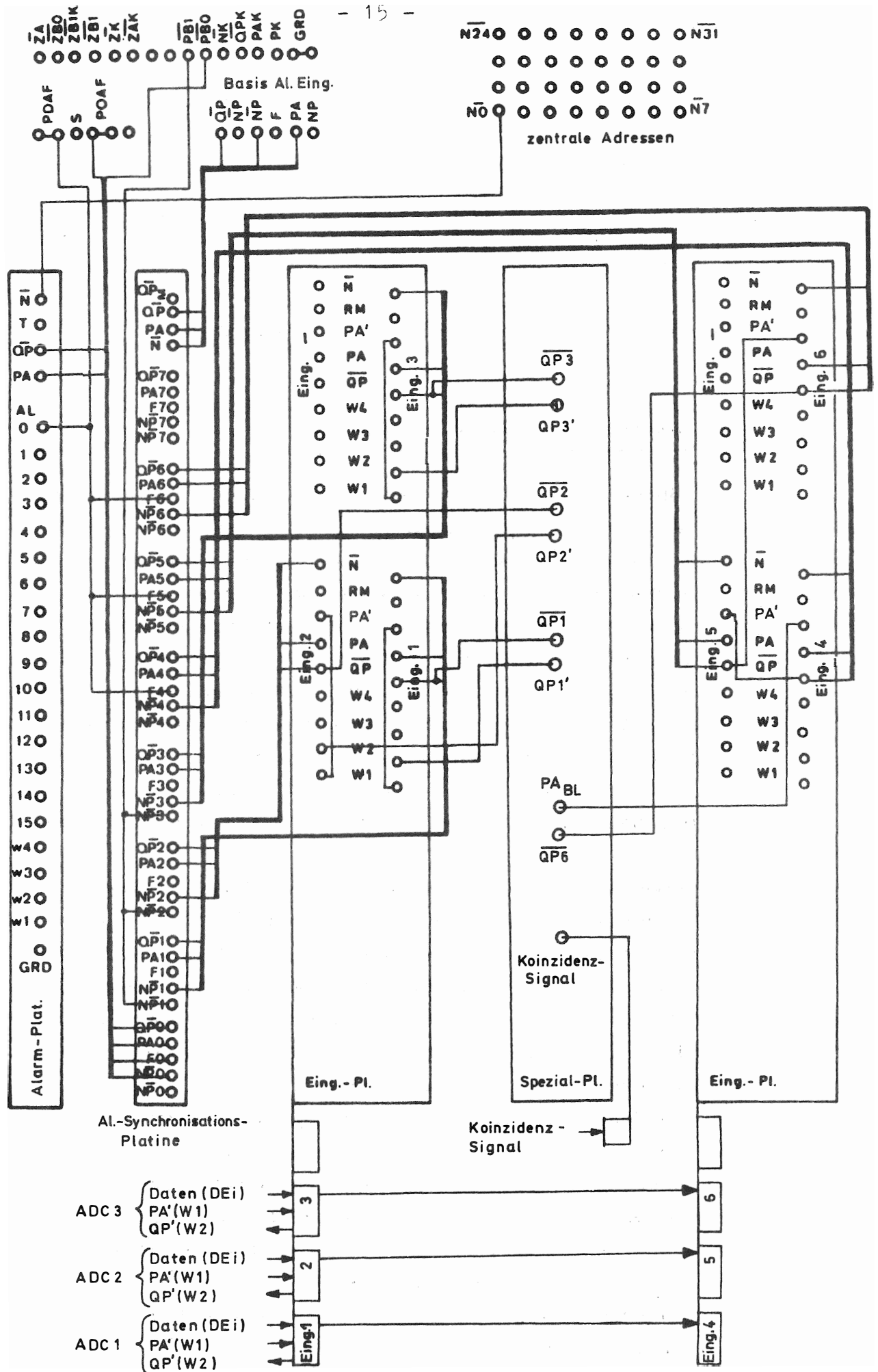


Abb. 3: Signalplan für die mehrparametrische Vielkanalanalyse