Zeitschrift: Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses

Herausgeber: Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen

Band: 71 (1980)

Heft: 11

Artikel: Conception de microprocesseurs spécialisés : exemple du microprocesseur horloger

Autor: Monbaron, J.-J. / Peguiron, N.

DOI: https://doi.org/10.5169/seals-905255

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Mehr erfahren

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. En savoir plus

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. Find out more

Download PDF: 13.12.2025

ETH-Bibliothek Zürich, E-Periodica, https://www.e-periodica.ch

Conception de microprocesseurs spécialisés: Exemple du microprocesseur horloger

Par J.-J. Monbaron et N. Peguiron

681.325–181.48:681.11.03;

1. Introduction

L'utilisation d'un microprocesseur pour réaliser une montre se justifie à plus d'un titre: d'une part, l'augmentation de la complexité des montres électroniques multifonctionnelles comportant par exemple un réveil, un chronographe, un timer, plusieurs fuseaux horaires, etc. a rendu la conception des circuits trop fastidieuse en logique câblée; d'autre part, bien que la montre soit un produit de consommation de masse, elle est vendue sous des formes très diverses tant du point de vue de l'habillage que du fonctionnement, ce qui implique pour le circuit électronique une grande versatilité [1].

Or, le recours à des microprocesseurs du commerce est impossible pour les raisons suivantes:

– nécessité avant tout de réduire la consommation (si possible moins que 1 µA en fonctionnement normal)

– réduction de la surface du circuit par une conception adaptée des entrées-sorties particulières de la montre.

L'intérêt du concept du microprocesseur pour la montre peut s'illustrer à l'exemple du *calcul de l'heure,* par la comparaison entre logique câblée et logique programmée:

Le circuit intégré traditionnel en logique câblée d'une montre électronique comporte principalement une chaîne de compteurs propres à chaque information horaire à traiter (fig. 1). Chacun d'entre eux assure deux fonctions, à savoir la mémorisation d'une information horaire et le traitement de cette information. Celui-ci comprend notamment l'initialisation (mise à l'heure), l'incrémentation et la détection de la valeur limite du compteur avec remise à 0 et report au compteur suivant.

Chacune de ces fonctions est assurée par un matériel distinct. La mémorisation est effectuée par des bascules de chaque compteur et le traitement par une logique combinatoire de programmation adaptée. Or, pour chaque compteur, la fonction mémoire est utilisée en permanence, alors que la logique de traitement travaille à une fréquence qui dépend de l'information horaire contenue dans le compteur. Pour la date, par exemple, la logique d'incrémentation n'est utilisée qu'une seule fois par jour. On peut parvenir à une utilisation plus rationnelle du matériel en adoptant une solution séquentielle, c'est-à-dire en réalisant les mêmes fonctions en logique programmée (fig. 1b). Dans cette nouvelle méthode de conception, la fonction mémoire est réalisée par une mémoire vive (RAM), de structure plus dense que les bascules des compteurs isolés. Quant aux circuits d'incrémentation et de comparaison, ils sont regroupés en un seul circuit arithmétique et logique (incrémenteur-comparateur) qui est utilisé pour le traitement successif de chaque information horaire (multiplexage de système). Les valeurs initiales et limites ainsi que le programme (fig. 2) sont stockés en mémoire morte (ROM), pour laquelle le prix de stockage d'un bit est environ trente fois inférieur au prix d'un bit mémorisé en logique câblée.

Cet exemple montre donc pourquoi on peut s'attendre à une réduction de la surface d'un circuit, en réalisant celui-ci en logique programmée.

Conférence de la Journée d'information de l'ASE sur «Le microprocesseur crée-t-il plus de problèmes qu'il n'en résout?», le 29 avril 1980 à Neuchâtel.

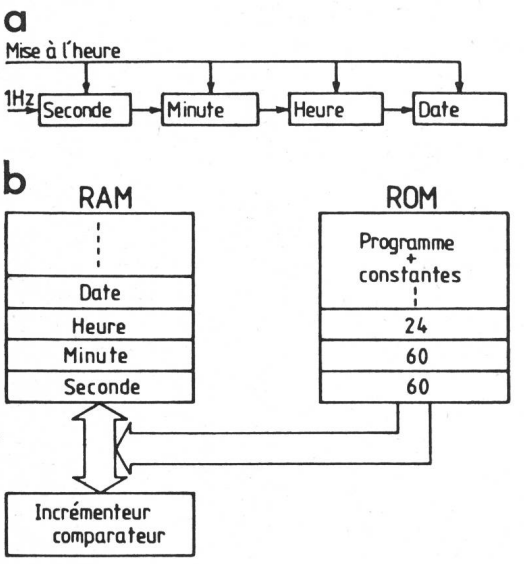

Fig. 1 Principe du processeur horloger
a) En logique cablée
b) En logique programmée
RAM Random Access Memory
ROM Read Only Memory

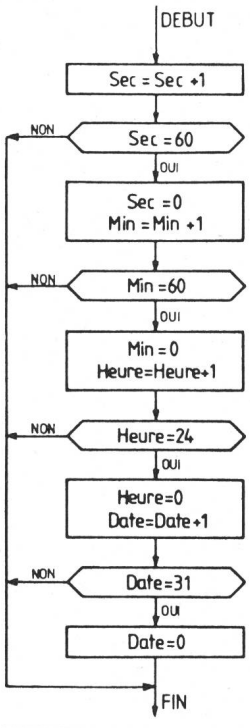

Fig. 2
Programme de calcul de l'heure correspondant à la fig. 1

2. Méthode de conception du processeur spécialisé

Le microprocesseur spécialisé ne se distingue pas d'un microprocesseur ordinaire par le schéma-bloc. Celui-ci contient en effet dans les deux cas, les mêmes éléments tels que RAM, ROM, ALU[1]), etc.

En revanche, le côté «spécialisé» du microprocesseur provient de la qualité de l'adaptation de la structure du microprocesseur au travail que celui-ci doit effectuer [4; 5; 6]. Cette adaptation s'effectue par le dimensionnement adéquat des paramètres suivants:

- format des données
- choix des instructions: nature et nombre, format des différents champs (opcode, constante, adresse...)
- système d'adressage: direct, indirect, immédiat et registres d'adresse associés
- logique d'interruption et de séquencement
- configuration des entrées-sorties (I/0)

Avant d'entreprendre l'optimisation simultanée de ces différents paramètres, il faut analyser en détail le cahier des charges d'une montre multifonctionnelle.

2.1 Cahier des charges

Le cahier des charges contient les fonctions typiques d'une montre multifonctionnelle suivantes:

- fonction montre heure/min/s ou heure/min/jour-date
- décompteur 24 h (timer)
- 1 réveil avec mélodie
- 1 fuseau horaire
- 1 chrono 100ième avec rattrapante

Afin de faire une première classification, on peut distinguer dans chacune de ces fonctions 4 types d'opérations principales qui sont: la commande manuelle (boutons-poussoirs), le calcul, la gestion des interruptions (start/stop du chrono, interruption de la seconde, du centième, etc.) et l'affichage.

1) Commande manuelle de la montre

Il ressort de l'organigramme de commande principal de la fig. 3 que:

– la montre peut occuper dans notre exemple, de manière continue ou momentanée, 25 modes différents quant à l'information affichée et à l'action des boutons-poussoirs associés.

– 54 chemins différents relient ces modes entre eux. Cependant, la complexité du mode d'emploi se réduit légèrement grâce à l'utilisation de la régularité de certaines décisions, par exemple, la fonction correction (voir fig. 3) a des parties communes aux fonctions montre, timer, réveil, fuseau.

2) Calcul associé à chaque fonction

Le problème du calcul lié à chaque fonction est simple. Il consiste, principalement à réaliser le comptage (ou décomptage) selon une base spécifique (60, 60, 24), voire variable.

De plus, la fonction réveil nécessite la comparaison à chaque minute de l'heure de la montre avec l'heure de réveil mémorisée.

3) Gestion des interruptions

Le temps de réaction de la montre à la pression d'un bouton-poussoir ne doit pas être perçu par l'utilisateur. Les fonctions du chrono doivent avoir une résolution de $^1/_{100}$ s. De plus, afin de réduire la consommation, le processeur doit être déclenché, une fois le traitement terminé, ce qui nécessite un

[1]) ALU Arithmetic and Logic Unit.

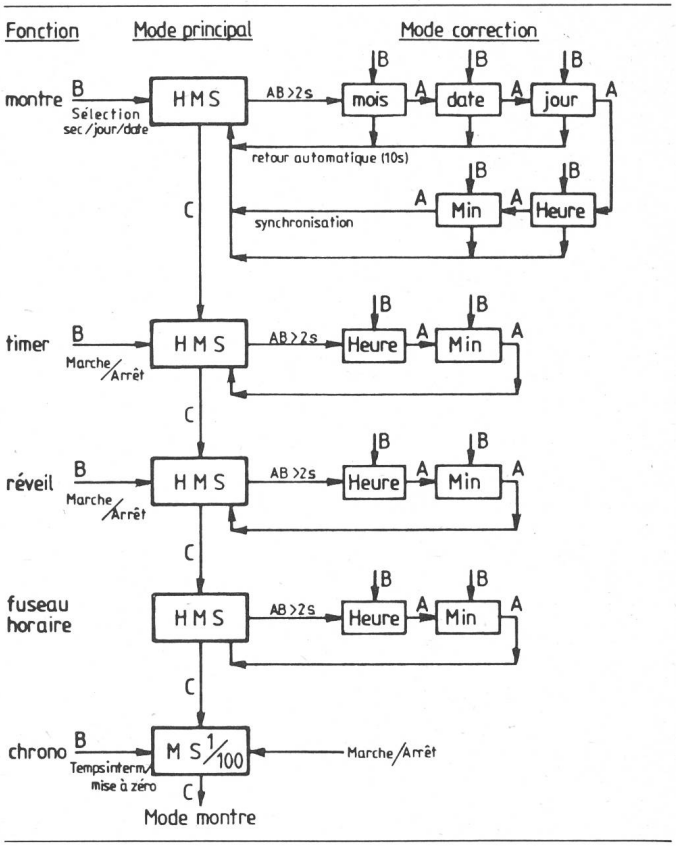

Fig. 3 Organigramme partiel de commande d'une montre
A, B, C Boutons poussoirs
HMS Affichage h, min, s
> 2 s Pression à maintenir > 2 s

dispositif d'enclenchement réagissant à la seconde de la montre, à $^1/_{100}$ s du chrono, ou à l'action sur un bouton-poussoir.

4) Affichage

L'affichage comporte dans notre exemple 53 segments LCD, organisés en 8 plages de fonction, 4 digits 7-segments, 2 digits alpha-numérique (8 et 9 segments) 2 bits de commande de la lumière et du transducteur électro-acoustique. Il est à relever que ces segments peuvent clignoter selon le mode.

5) Commentaire

De l'analyse du cahier des charges, il ressort que la complexité de la commande et de l'affichage dépasse de loin celle du calcul. Cependant, en mode de fonctionnement normal, les boutons-poussoirs ne sont pas utilisés; en effet, dans la majorité des cas, le processeur ne fait que incrémenter la seconde et afficher la nouvelle valeur.

2.2 Discussion de l'architecture du processeur

Le premier argument important pour la conception de la machine est représenté par le rapport de la fréquence d'horloge de base de la machine (16 kHz) à la fréquence de travail des fonctions rapides (chrono et musique). En effet, ces dernières doivent être servies à $^1/_{100}$ s, ce qui représente 160 impulsions d'horloge ou environ 50 instructions à raison de 3 phases par instruction.

Cette contrainte impose naturellement de diviser le programme en une routine de calcul rapide desservie par interrup-

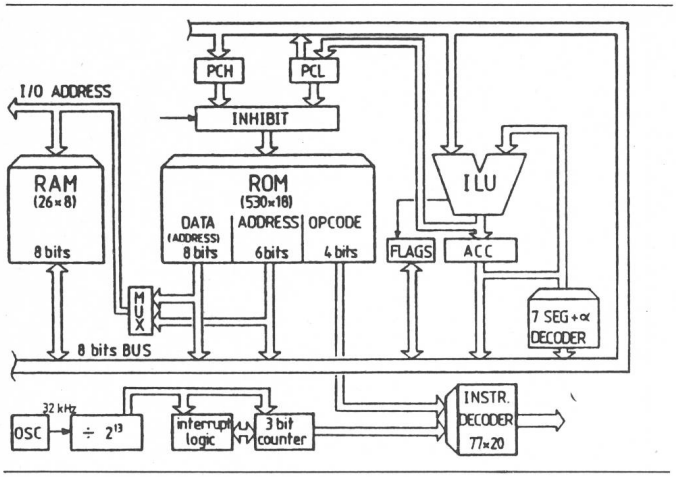

Fig. 4 Architecture du processeur horloger

tion (à 1 niveau) et un module principal de commande et d'affichage.

Lors de chaque interruption, le compteur de programme (PC) est sauvegardé dans une position fixe de la RAM, ce qui évite l'introduction et la manipulation d'un pointeur de pile (SP). De plus, le temps pris par la routine d'interruption doit être suffisament inférieur à $^1/_{100}$ s afin de permettre également le traitement des autres fonctions permanentes (calcul de l'heure, du timer, etc.) et occasionnelles (boutons-poussoirs), ce qui implique un set d'instructions puissantes et rapides.

Parallèlement, on désire réaliser une machine capable d'effectuer ces instructions puissantes avec une surface minimale. Pour ce faire, une architecture ne comprenant ni registre d'adresse de la RAM, ni registre d'instruction a été développée (fig. 4).

Une telle structure impose un format d'instruction comprenant en parallèle le code opératoire (4 bits), l'adresse de l'opérande (6 bits) et un champ de données (8 bits) pour la spécification des constantes (valeur d'overflow, adresse de saut, adresse du 2e opérande, etc.). Le format des variables (RAM), est de 2×4 bits BCD bien adapté à l'information horaire. Ce calcul (incrémentation et décrémentation BCD, etc.) s'effectue dans une unité de traitement appropriée appelée ILU (Increment Logic Unit), connectée de manière à permettre l'incrémentation du PC_L (8 bits inférieurs) en une seule impulsion d'horloge.

2.3 Set d'instructions et séquencement

Le set d'instructions (table I) comprend 19 instructions, dont 8 sont très puissantes et spécialisées. Par exemple, l'instruction «INR SEC 59» compare la valeur de la RAM adressée par SEC avec 59 et effectue la remise à 0 en cas d'égalité. Dans le cas contraire, elle effectue l'incrémentation BCD. Le séquencement de cette instruction dure 4 ou 6 impulsions d'horloge (table II).

L'instruction «JPB ⌀10 (valeur) 011 (masque) LABEL» compare les 3 flags A, B et C à travers les «masques» 011 avec les valeurs correspondantes ⌀10 (⌀ état indifférent) et effectue le saut conditionnel à l'adresse LABEL, en 2 impulsions d'horloge:

0 comparaison de ⌀10 avec ABC
1 PC ← PC + 1 ou LABEL

L'instruction JPC permet d'effectuer l'opération analogue sur 3 autres flags.

Table I

N°	Nom	Adresse destination	Adresse source data (8)	Transfert	Commentaire
0	MVI	A	Data	RAM (A) ← Data	Initialisation
1	XRI	A	Data	RAM (A) ← RAM (A) ⊗ Data	EX-OR immédiat
2	ANI	A	Data	RAM (A) ← RAM (A) AND Data	AND immédiat
3	ORI	A	Data	RAM (A) ← RAM (A) + Data	OR immédiat
4	MOV	A	B	RAM (A) ← RAM (B)	Transfert direct-direct
5	XOR	A	B	RAM (A) ← RAM (B) ⊗ RAM (A)	EX-OR direct-direct
6	INB	A	–	RAM (A) ← RAM (A) + 1	incrémentation directe (sans overflow)
7	DCB	A	–	RAM (A) ← RAM (A) − 1	décrémentation directe (sans overflow)
8	CMP	A	B	EQ ← 1 si RAM (A) = RAM (B)	Comparaison directe-directe
9	CLR	A	–	RAM (A) ← 0	Clear double
10	AFF	A	B	RAM (A) $\overset{\text{7-seg. dec}}{\leftarrow}$ RAM (B)	Décodage numérique
11	ALF	A	B	RAM (A) $\overset{\alpha\text{-dec}}{\leftarrow}$ RAM (B)	Décodage alpha-numérique
12	RET	A	Mode*	PC ← RAM (B)	Saut indirect (RETURN)
13	HALT	–	Mode*	–	HALT
14	JPB	A_1, A_2	B	PC ← B si ($flag_L + A_1$) + A_2	Test de flag + saut conditionel
15	JPC	A_1, A_2	B	PC ← B si ($flag_H + A_1$) + A_2	Test de flag + saut conditionel
16	CALL	A	B	RAM (A) ← PC PC ← B	CALL avec sauvegarde du PC dir.
17	CPI	A	B	EQ ← 1 si RAM (A) = B	Comparaison immédiate
18	INR	A	B	RAM (A) $\overset{\text{BCD}}{\leftarrow}$ RAM (A) + 1	Incrémentation BCD avec test d'overflow + reset
19	DCR	A	B	RAM (A) $\overset{\text{BCD}}{\leftarrow}$ RAM (A) − 1	Décrémentation BCD avec test d'overflow + reset

L'affichage est effectué au moyen de 2 instructions spéciales, par exemple «AFF SEC DIGIT 0» séquencée comme suit:

0 ACC ← RAM (SEC)
1 RAM (DIGIT 0) $\xleftarrow{\text{7-seg}}$ ACC_{Hi}
2 RAM (DIGIT 0 + 1) $\xleftarrow{\text{7-seg}}$ ACC_{Lo}
3 PC ← PC + 1

On constate que le décodage est effectué par une logique appartenant à l'ILU; d'autre part les registres de sortie (Output latches) sont assimilés par la méthode d'adressage à une position de la RAM, ce qui simplifie le séquencement de l'opération. Ainsi, en 4 impulsions d'horloge, il est possible d'afficher 2 digits.

3. Réalisation des programmes

Le caractère direct et parallèle de l'architecture se retrouve dans l'écriture très linéaire des programmes. Ceux-ci contiennent très peu d'instructions de «gestion de programme» (appel de sous-routine, table de branchement, manipulation de pointeurs, etc.) par rapport aux instructions «utiles» qui traitent effectivement l'information horaire et les entrées-sorties.

Ce fait s'illustre par un exemple de programme de calcul de l'heure:

```
DEBUT:      INR  SEC   59H;     Incrémentation secondes
            JPC  CYO   FIN;     fin si pas de report
            INR  MIN   59H;     incrémentation minutes
            JPC  CYO   ALTST;   aller tester le réveil si pas
                                de report
            INR  HR    23H;     incrémentation heures
            JPC  CYO   ALTST;   ...
              .
              .
              .
ALTST:      CMP  MIN   ALMIN;   comparaison minutes
                                avec minute-réveil
            JPC  EQO   FIN;     fin si différent
            CMP  HR    ALHR;    comparaison heures...
            JPC  EQO   FIN;     ...
SET ALARM:  ...
FIN:        RET  PC SAVE 1;     retour à l'adresse sauvée
                                en PCSAVE... (pas de
                                RET conditionnel)
```

La commande des fonctions de la montre contient beaucoup de tests et de branchements. Ceux-ci (qui représentent la plus grande partie du programme) peuvent être effectués de manière efficace par les instructions de branchement spécialisées:

```
TSTBPS:  MOV  FLAGS  BPS;       val. boutons poussoir
                                → Flags
         JPB  CO A1  TST1;      si C = 0 et A = 1
                                (indépendemment de B!)
                                aller à TST1
         JPB  CO B1  TST2;      si C = 0 et B = 1,
                                aller à TST2
         JPB  C1 BO AO RTNC;    si C = 1...
         RET  PCSAVE 1;         tous les autres cas ne sont
                                pas considérés...
TST1:    JPB  B1 RTNAB;         si B = 1 aller à RTNAB
RTNA:    .
         .
         .
TST2:    JPB  A1 RTNAB;         si A = 1 aller à RTNAB
RTNB:    .
         .
         .
RTNAB:   .
         .
         .
```

Table II

Sans overflow	Avec overflow
0 ACC ← RAM (SEC)	0 ACC ← RAM (SEC)
1 Carry ← 0 car ACC ≠ 59	1 Carry ← 1 car ACC = 59
2 ACC ← RAM (SEC) + 1	2 ACC ← 0
3 ACC ← ACC ajusté décimal	3 RAM (SEC) ← ACC
4 RAM (SEC) ← ACC	4 PC ← PC + 1
5 PC ← PC + 1	

L'exemple ci-dessus est le programme équivalent à un arbre de décision qui a la propriété de décoder les cas RTNA, RTNB, RTNC et RTNAB correspondant aux modes d'actions permis des boutons-poussoirs A, B et C en 2 instructions de test par cas (3 pour RTNC).

En conclusion, le programme est articulé en un module de commande et d'affichage pour chaque fonction de la montre, dotés de un niveau de sous-routines communes qui sert à la fois au service de l'interruption de la seconde (calcul de l'heure) et à l'exploitation judicieuse des rares répétitions de l'organigramme de cahier des charges. Un deuxième niveau, indépendant du premier et prioritaire, permet le calcul du chronographe. Seule la sauvegarde des «flags» est nécessaire pour rendre cette routine invisible au reste du programme. Ceci est dû au fait que le langage de la machine ne fait aucune référence explicite à des registres de travail (accumulateur, registres d'adresse), à l'exception du registre des «flags». Ainsi, les instructions logiques et d'incrémentation agissent, pour le programmeur, indifféremment sur toutes les positions de la RAM et des I/0.

4. Conclusions

Les résultats de la conception du microprocesseur horloger décrit ci-dessus sont les suivants:

ROM (programmes)	530 × 18 bits
RAM (données)	26 × 8 bits
OUTPUT latches + flags + ...	16 × 8 bits
PLA (séquenceur)	77 × 20 bits
PLA (décodeurs)	10 × 7; 42 × 4; 15 × 8 bits
ILU	700 transistors MOS
INTERRUPT LOGIC (REQUEST et PRIORITY):	180 transistors MOS

Dans l'état actuel de la technologie C-MOS, on peut estimer la surface du circuit à 17 mm² à l'exclusion du Bus, des drivers de sortie, et de la chaîne de division.

Ces résultats montrent l'intérêt d'une réalisation spécialisée d'un microprocesseur, tant du point de vue de la surface que de la consommation. En effet, bien qu'il ne soit pas possible d'estimer la valeur de cette dernière sans réaliser le circuit, on peut s'attendre à une bonne performance en considérant le très faible nombre d'instructions de service par rapport aux instructions de traitement effectif de l'information.

De même, le bon rapport entre les bits d'opcode (4) et les bits de données (14) conduit à une faible surface de la mémoire de programme (ROM), tout en offrant par sa structure parallèle, une excellente rapidité d'exécution.

L'étude de ce processeur spécialisé représente la 3e génération de processeurs horlogers développés dans notre Institut [4; 6; 7], dans le cadre d'un groupe de travail sur les microprocesseurs financé par la «Commission d'Encouragement pour la Recherche Scientifique» (Projet N° 988).

Cette étude a permis d'illustrer une méthode de conception d'un microprocesseur spécialisé dans laquelle la conception du

«software» précède la conception du «hardware». Ceci oblige le logicien concepteur de circuits à raisonner en informaticien pour analyser le problème, puis, à établir le set d'instructions le mieux adapté à l'organigramme obtenu par l'analyse et, enfin, à dessiner le système.

Adresse des auteurs

Dr. *J. J. Monbaron* et *N. Peguiron*, Institut de microtechnique de l'Université de Neuchâtel, Maladière 71, 2000 Neuchâtel.

Bibliographie

[1] *J.-F. Perotto* et *C. Piguet:* Qu'est-ce qu'un processeur? Bulletin de la Société Suisse de la Chronométrie 7(1977)3, p. 305...308.

[2] *D. Mange:* Arbres de décision pour systèmes logiques câblés ou programmés. Bull. ASE/UCS 69(1978)22, p. 1238...1243.

[3] *C. Piguet* e.a.: Le séquenceur d'un microprocesseur. Bull. ASE/UCS 70(1979)3, p. 126...132.

[4] *D. Gigon:* Réalisation d'une horloge avec une structure de microprocesseur. Travail de diplôme de l'Institut de Microtechnique, Université de Neuchâtel, 1979.

[5] *C. Piquet, J.-F. Perotto* et *J.-J. Monbaron:* Conception d'un microprocesseur horloger. Proceedings du Congrès International de Chronométrie, Genève, 1979, conference E 2.1, p. 271...278.

[6] *J.-J. Monbaron* e.a.: Conception d'un microprocesseur horloger. Journée d'Electronique, Lausanne 1979, p. F3–1...F3–10.

[7] *J.-J. Max* et *J.-J. Monbaron:* Evaluation d'un microprocesseur horloger basé sur un format de 4 bits. Rapport interne de l'Institut de Microtechnique, Université de Neuchâtel, 1979.

[8] *N. Peguiron:* Cross assembleur universel pour microprocesseurs specialisés. Rapport interne 39 EC 03005 de l'Institut de Microtechnique, Université de Neuchâtel, 1980.

Julius Joh. Ph. L. Elster 1854–1920 Hans Friedrich Geitel 1855–1923

Julius Joh. Ph. L. Elster

Bildquelle: Deutsches Museum München

Am Weihnachtstag 1854 erblickte Julius Elster in Blankenburg im Harz das Licht der Welt. Sein Vater war Theologe und Privatlehrer. Julius studierte Physik in Berlin und Heidelberg, doktorierte 1879 und wurde an Ostern 1881 Lehrer am Gymnasium Wolfenbüttel. Diesen Posten behielt er bis zu seiner Pensionierung im Jahre 1919. Er heiratete die Tochter eines Bankiers, mit der er in harmonischer aber kinderloser Ehe lebte. Wenige Tage nach seinem Tod schied seine Frau aus Gram freiwillig aus dem Leben.

Das gastliche Haus, das das Paar bewohnte, bildete das geistige Zentrum der Stadt. Es beherbergte auch Hans Geitel, einen Jugendfreund Elsters, der ebenfalls Physiklehrer am Gymnasium geworden war. Dieser war praktisch an allen Forschungen beteiligt, die Elster im Laboratorium des Privathauses machte.

Die beiden begannen ihre Forschertätigkeit 1884 mit ganz einfachen Mitteln. Ihr Interesse galt zuerst der Gewitterelektrizität und dem elektrischen Feld der Erde. Von 1889 an beschäftigten sie sich mit lichtelektrischen Phänomenen. Innert 3 Jahren gelang ihnen die Konstruktion einer Photozelle, die so empfindlich war, dass sie auf das Licht einer Kerze in einer Entfernung von einem Kilometer reagierte. Sie bestand aus einem evakuierten Glasrohr mit zwei Elektroden. Die aus verschiedenen Alkalimetallen bestehende Kathode konnte von aussen belichtet werden; wurde dann Spannung angelegt, so entstand im Gas ein Strom, der sich proportional zum eingestrahlten Licht veränderte. Die Erfindung dieser Photozelle ist eines der Hauptverdienste der beiden Forscher. Damit machten sie auch astronomische Beobachtungen und massen bei verschiedenen Sonnen- und Mondfinsternissen den zeitlichen Verlauf der Lichtstärke.

In den Jahren vor der Jahrhundertwende untersuchten die beiden die Leitfähigkeit der Luft. Sie stellten dabei fest, dass diese von der Menge der vorhandenen freien Ionen abhängt, die ihrerseits, wie sie vermuteten, von der Radioaktivität herrührt und wahrscheinlich kosmischen Ursprungs ist. Sie prägten 1899 den Begriff «Atomenergie», also Energie, die aus dem Atom stammt.

Die beiden Forscher berichteten in mehr als 100 wissenschaftlichen Mitteilungen über die Ergebnisse ihrer Arbeiten, die sie übrigens ganz aus eigenen Mitteln bestritten. Verschiedene wissenschaftliche Vereine ernannten sie zu Ehrenmitgliedern. Am 8. April 1920 starb Elster an Zuckerkrankheit; sein Freund Geitel überlebte ihn um 3 Jahre. *H. Wüger*