United States Patent [19]

Stambera et al.

[11] **3,851,151**

[45] **Nov. 26, 1974**

[54] **WEIGHING AND PRICE CALCULATING DEVICE**

[76] Inventors: **Adolf Stambera,** Dinkelsbuhlstr. 11, Stuttgart-Bad Cannstatt; **Eberhard Schmid,** Beethovenstrasse 14, Schwieberdingen, both of Germany

[22] Filed: **Dec. 14, 1972**

[21] Appl. No.: **315,120**

Related U.S. Application Data

[63] Continuation-in-part of Ser. No. 87,570, Nov. 6, 1970, abandoned.

[52] **U.S. Cl.** ..... **235/61.9 R,** 177/26, 235/61 PS, 235/151.33

[51] **Int. Cl.**.. **G06k 15/02,** G01g 23/16, G06f 15/20

[58] **Field of Search** ..... 235/61.9 R, 61.7 R, 61.6 R, 235/61 PS, 151.33; 177/13, 5, 25, 26

[56] **References Cited**

UNITED STATES PATENTS

2,974,863	3/1961	Williams, Jr. et al.	235/151.33
3,039,686	6/1962	Bell et al.	235/151.33
3,393,302	7/1968	Cichanowicz et al.	235/151.33
3,557,353	1/1971	Allen et al.	235/151.33
3,635,297	1/1972	Salava	235/151.33

Primary Examiner—Daryl W. Cook

[57] **ABSTRACT**

A device for weighing commodities, for automatically calculating the price of the weighed commodity on the basis of a tariff, and for printing out the end price and weight on a ticket to accompany the commodity. The device is equipped for a quick and foolproof adaption to a change of commodity, and by a special combination of memories and code selector switches provides a versatile instrument for rounding up calculated values, to conversion to different currency systems, for summing and storing selected signals, and for modifying the weight/price tariff.

5 Claims, 29 Drawing Figures

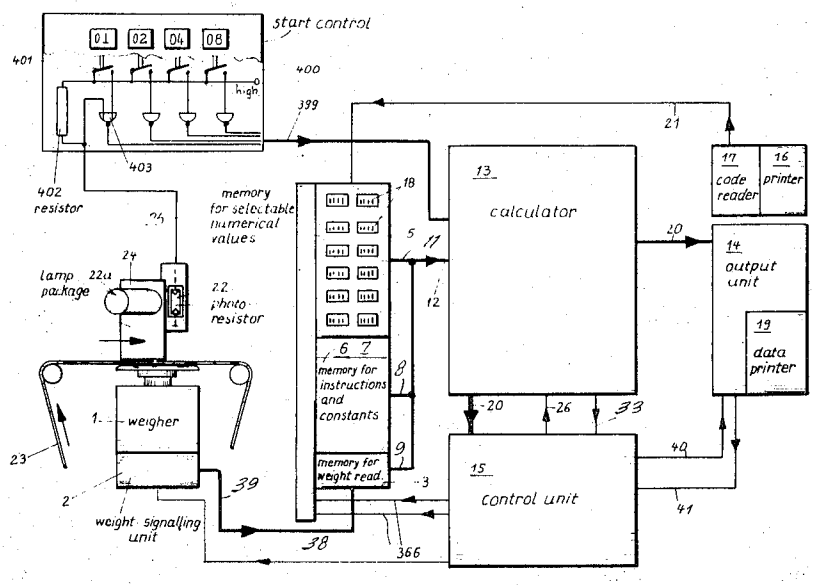


Fig. 1

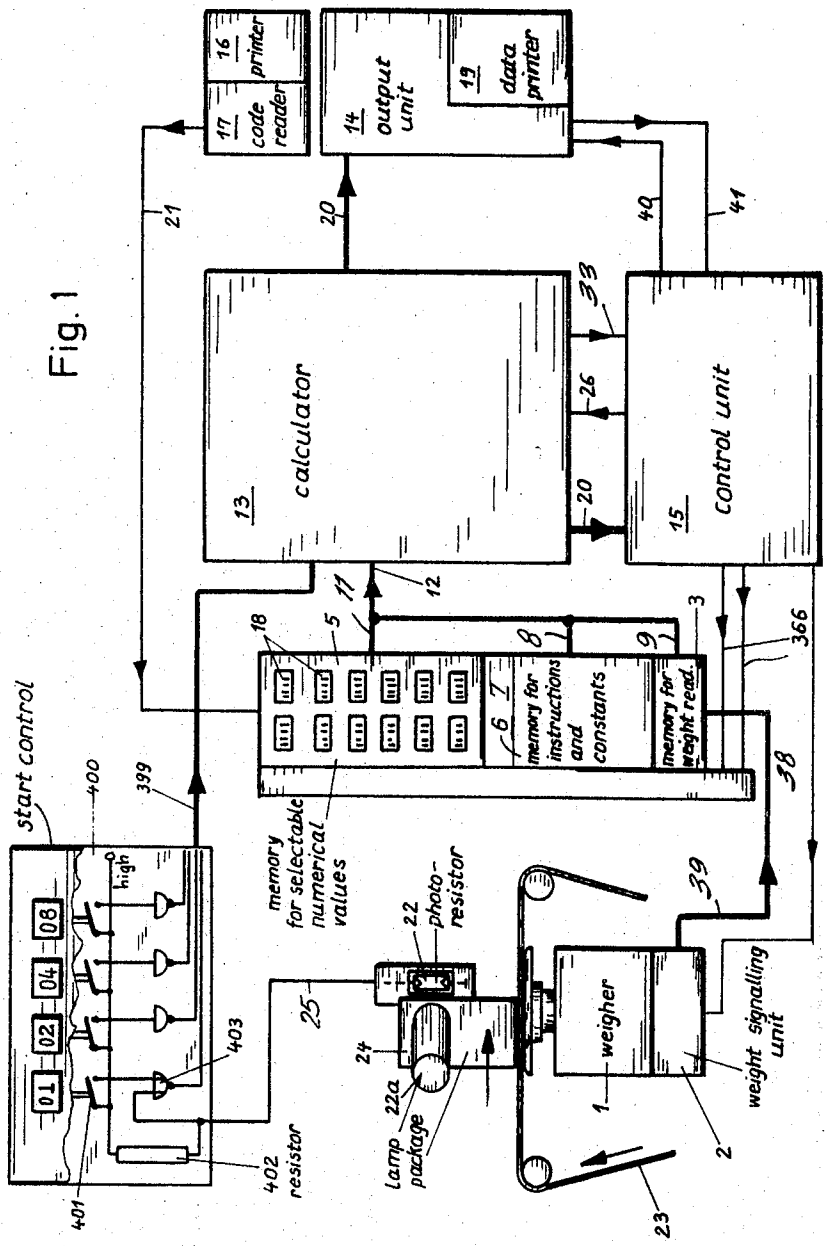

16 printer
17 code reader
14 output unit
19 data printer
21
20
40
41
13 calculator
33
26
20
15 control unit
11
12
5
18
8
9
3
366
6 7 memory for instructions and constants
memory for weight read.
38
start control
400
399
high
01
02
04
08
403
402 resistor
401
memory for selectable numerical values
22 photo-resistor
25
24
22a Lamp package
39
weigher
1
2
weight signalling unit
23

Fig. 2

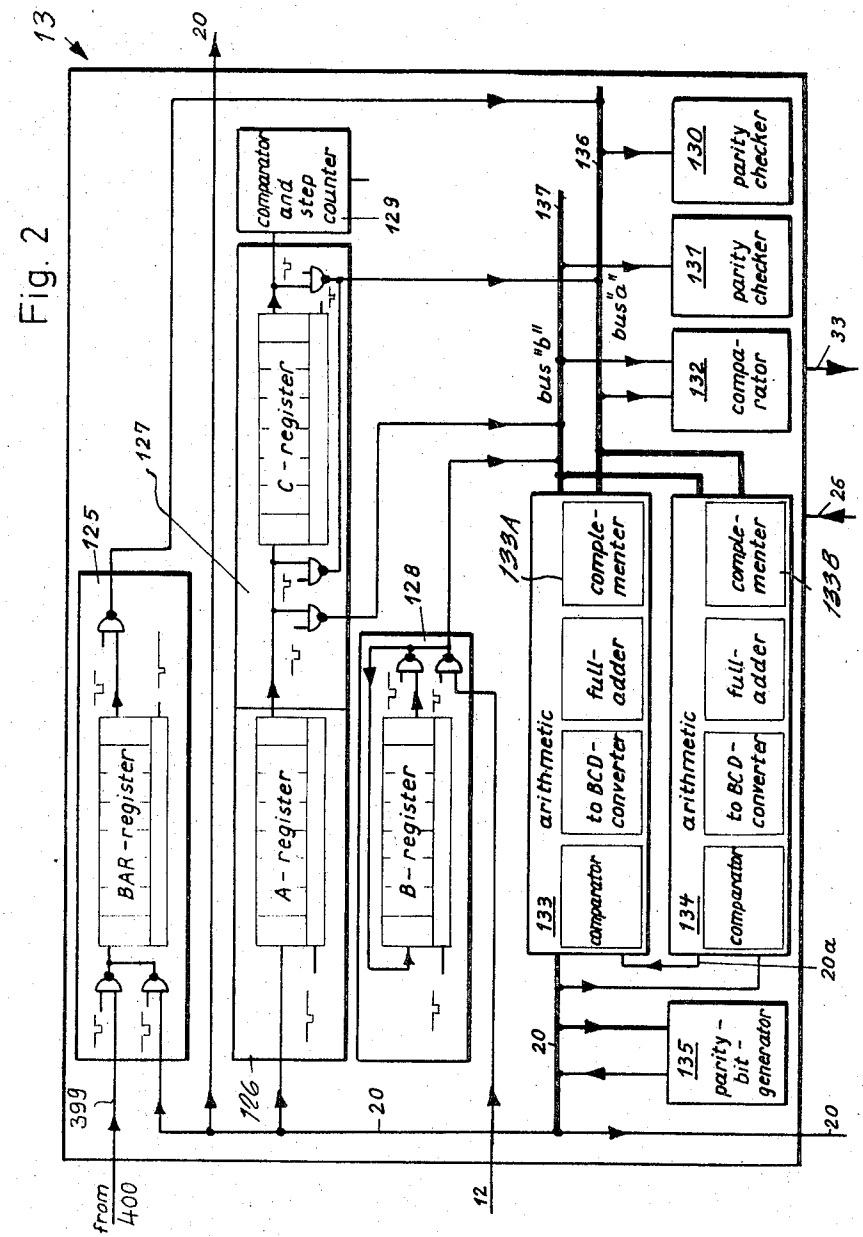
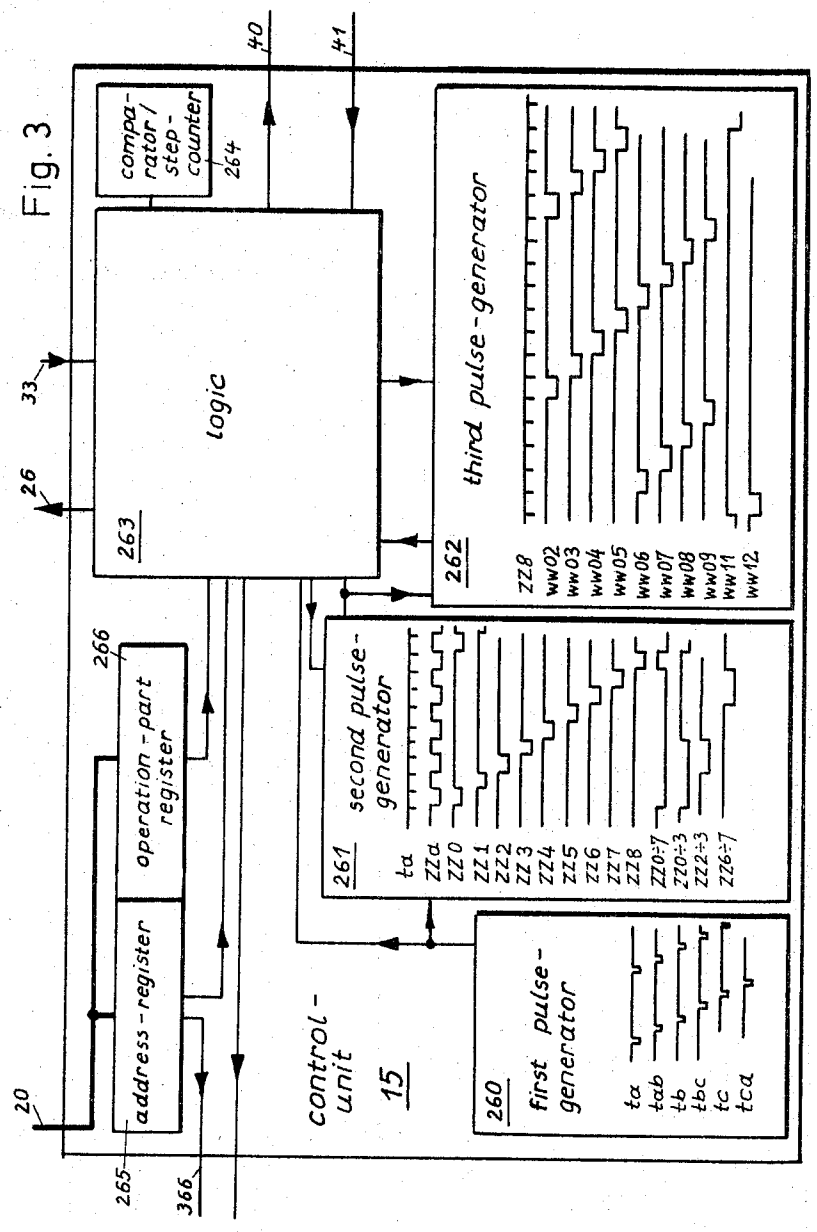

13
20
comparator and step counter
129
C - register
127
125
BAR - register
A - register
B - register
128
126
399
from 400
20
12
bus "b"
bus "a"
136
137
130 parity checker
131 parity checker
132 compa-rator
33
26
133A
133B
133 arithmetic
134 arithmetic
comparator
to BCD-converter
full-adder
comple-menter
135 parity-bit-generator
20a

Fig. 3


40
41
comparator/step-counter
264
logic
263
33
26
third pulse-generator
262
ZZ8
ww02
ww03
ww04
ww05
ww06
ww07
ww08
ww09
ww11
ww12
second pulse-generator
261
ta
ZZa
ZZ0
ZZ1
ZZ2
ZZ3
ZZ4
ZZ5
ZZ6
ZZ7
ZZ8
ZZ0÷7
ZZ0÷3
ZZ2÷3
ZZ6÷7
first pulse-generator
260
ta
tab
tb
tbc
tc
tca
control-unit
15
operation-part register
266
address-register
265
366
20

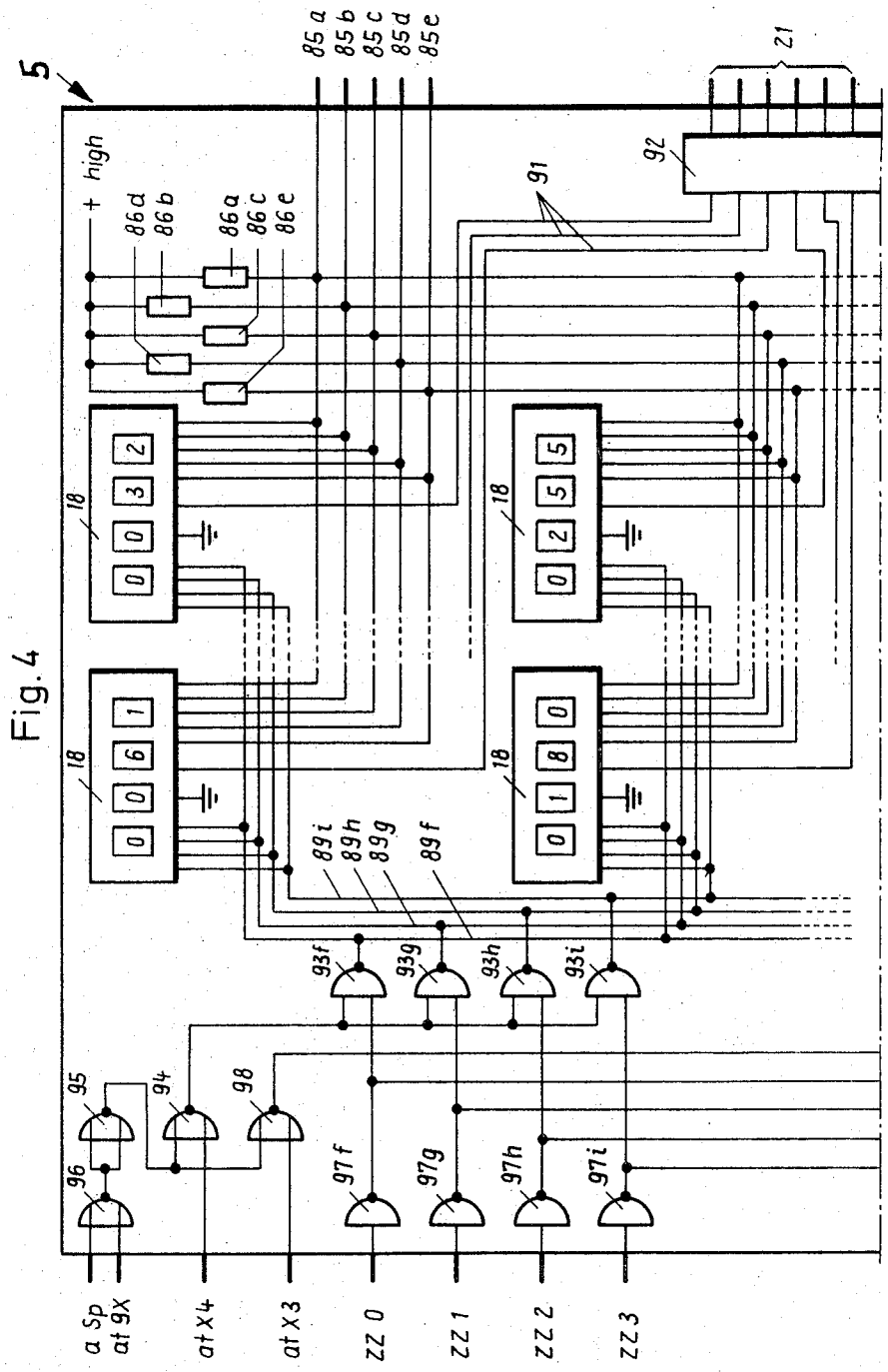

5
85a
85b
85c
85d
85e
21
92
91
+ high
86d
86b
86a
86c
86e
18
89i
89h
89g
89f
93f
93g
93h
93i
95
94
98
96
97f
97g
97h
97i
a Sp
at gX
at X4
at X3
zz 0
zz 1
zz 2
zz 3

Fig. 4

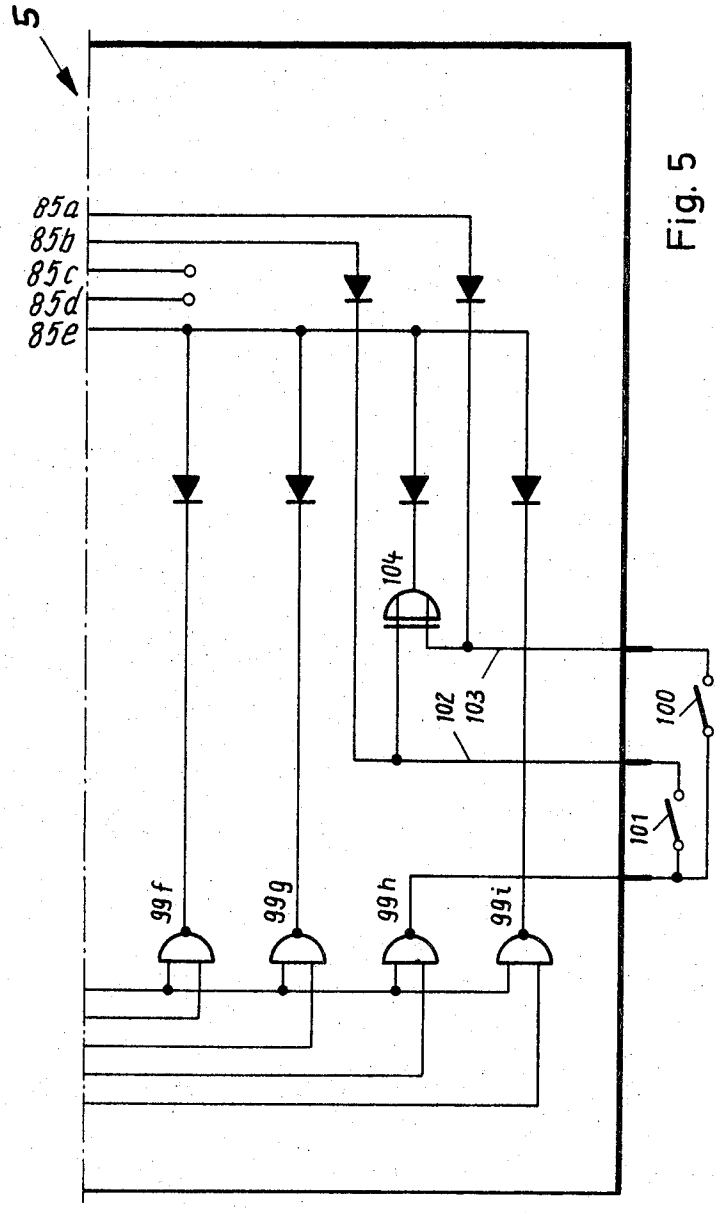


Fig. 5

Fig. 6

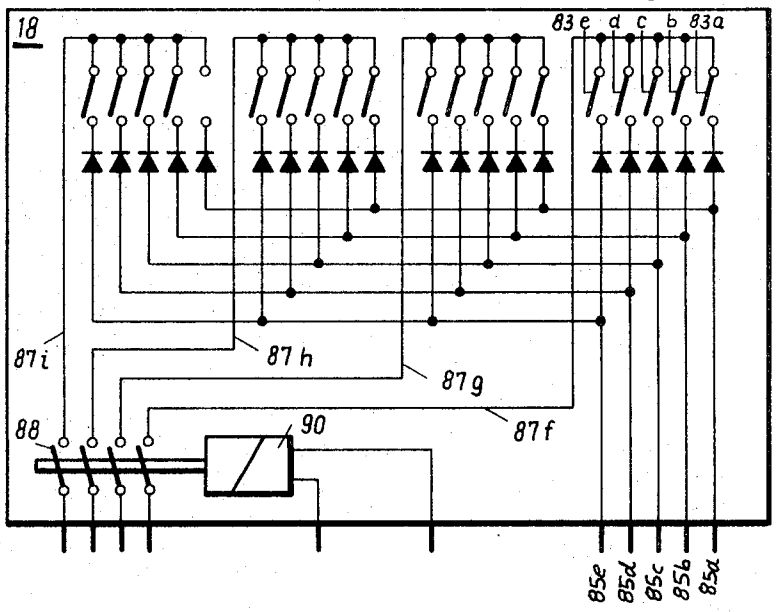

18
83e d c b 83a
87i
87h
87g
87f
88
90
85e 85d 85c 85b 85a

Fig. 7

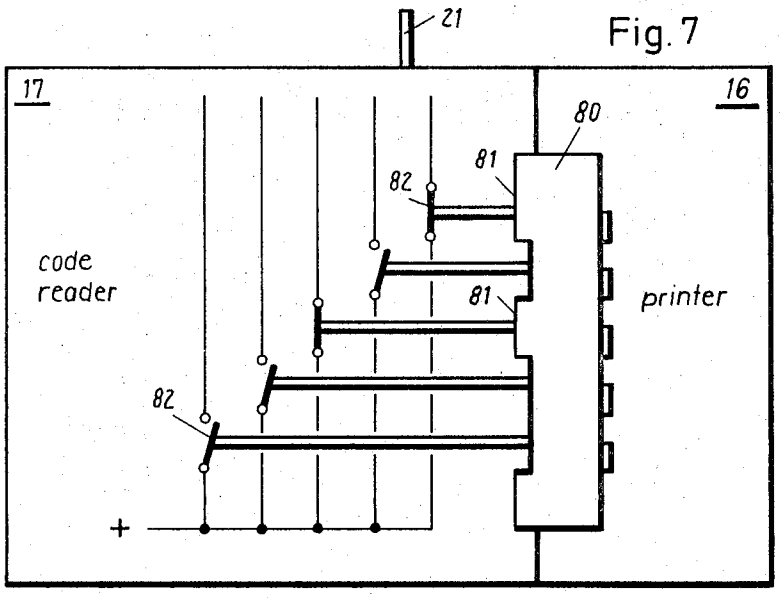
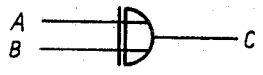

21
17
16
80
81
82
code reader
printer
+

Fig. 8

exclusive - or - gate



input A	input B	output C
0	0	0
0	1	1
1	0	1
1	1	0

Fig. 19

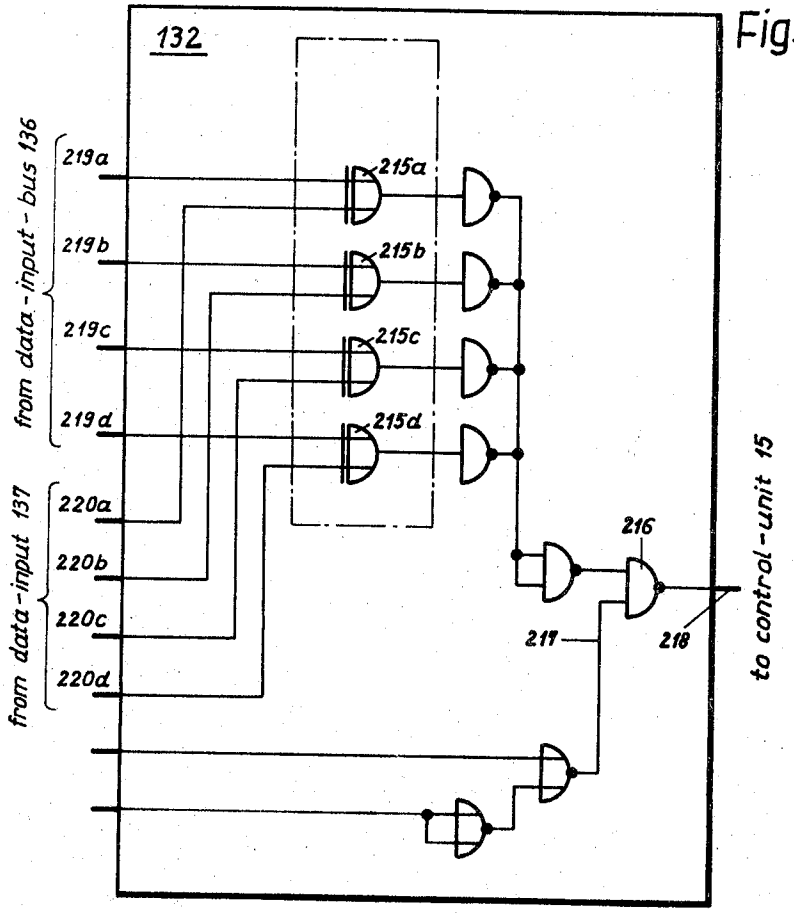


Fig.9

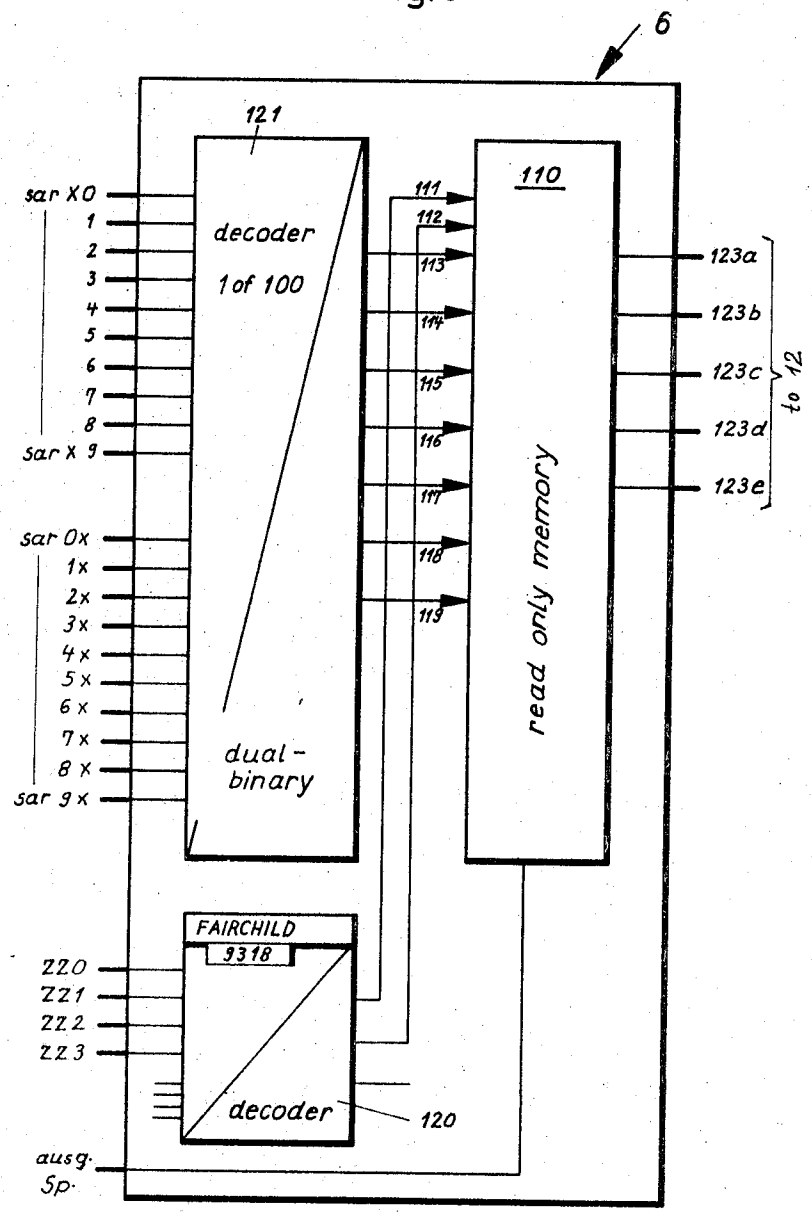
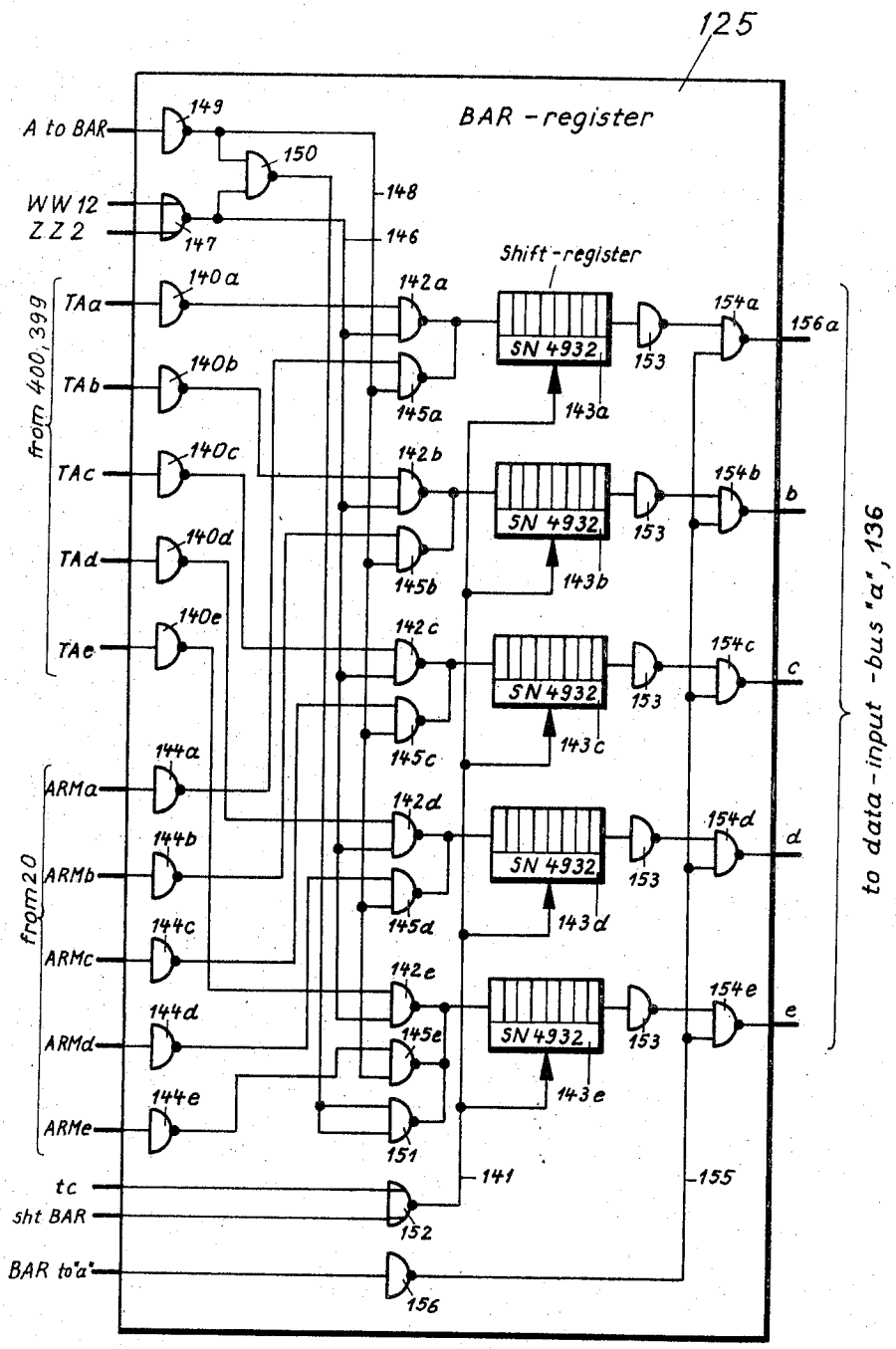

6
121
decoder
1 of 100
dual-
binary
110
read only memory
sar X0
1
2
3
4
5
6
7
8
sar X 9
sar 0x
1x
2x
3x
4x
5x
6x
7x
8x
sar 9x
111
112
113
114
115
116
117
118
119
123a
123b
123c
123d
123e
to 12
FAIRCHILD
9318
ZZ0
ZZ1
ZZ2
ZZ3
decoder
120
ausg.
Sp.

Fig. 10


125
BAR - register
A to BAR
WW 12
ZZ 2
149
150
147
148
146
Shift - register
from 400, 399
TAa
TAb
TAc
TAd
TAe
140a
140b
140c
140d
140e
142a
142b
142c
142d
142e
145a
145b
145c
145d
145e
SN 4932
143a
143b
143c
143d
143e
153
154a
154b
154c
154d
154e
156a
b
c
d
e
to data - input - bus "a", 136
from 20
ARMa
ARMb
ARMc
ARMd
ARMe
144a
144b
144c
144d
144e
151
141
155
tc
sht BAR
152
BAR to "a"
156

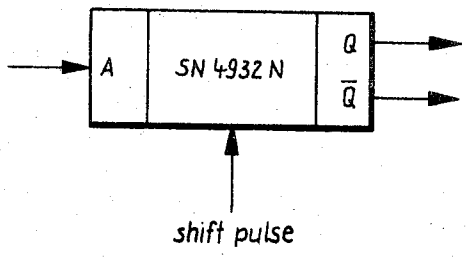


Fig. 11

truth-table for SN 4932 N 8-bit-shift-register		
tn	tn + 8	
A	Q	$\overline{Q}$
0	0	1
1	1	0

Fig. 12

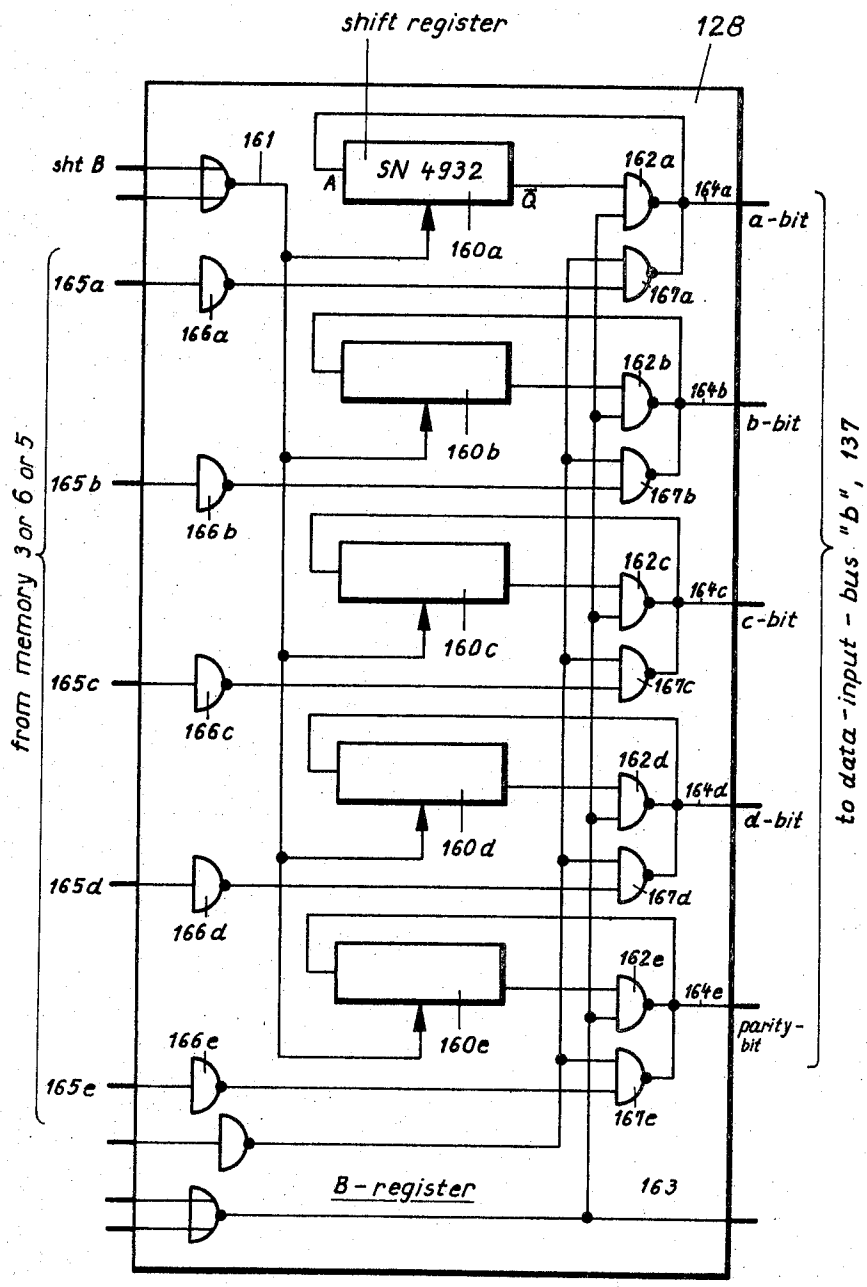

shift register
128
sht B
161
SN 4932
A
Q
160a
162a
164a
a-bit
165a
166a
167a
160b
162b
164b
b-bit
165b
166b
167b
160c
162c
164c
c-bit
165c
166c
167c
160d
162d
164d
d-bit
165d
166d
167d
160e
162e
164e
parity-bit
165e
166e
167e
from memory 3 or 6 or 5
to data-input-bus "b", 137
B-register
163

Fig. 13

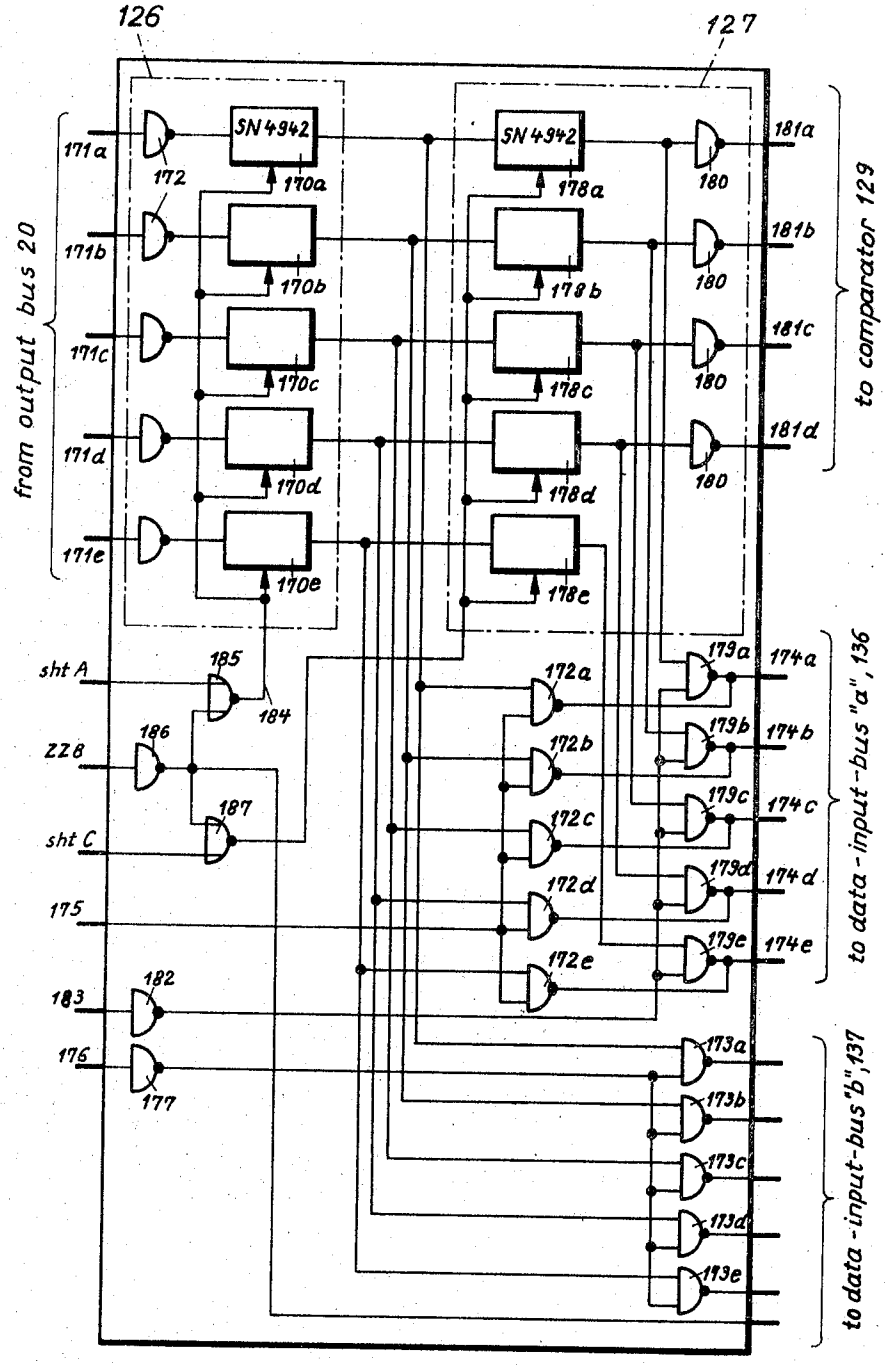

126
127
SN4942
SN4942
from output bus 20
171a
171b
171c
171d
171e
172
170a
170b
170c
170d
170e
178a
178b
178c
178d
178e
180
181a
181b
181c
181d
to comparator 129
sht A
ZZB
sht C
175
183
176
185
184
186
187
182
177
172a
172b
172c
172d
172e
173a
173b
173c
173d
173e
174a
174b
174c
174d
174e
to data-input-bus "a", 136
to data-input-bus "b", 137

Fig. 14

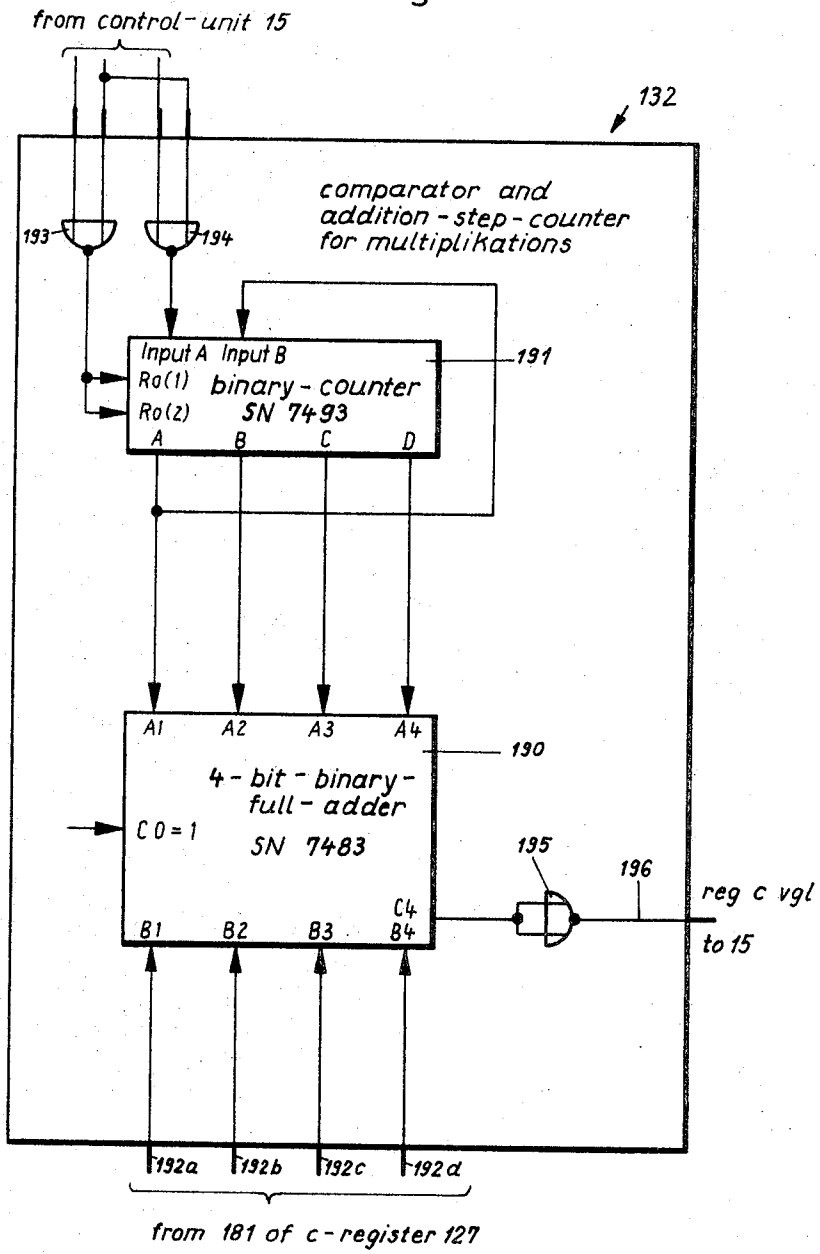

from control-unit 15
132
comparator and addition-step-counter for multiplikations
193
194
Input A Input B
Ro(1)
Ro(2)
binary-counter SN 7493
A B C D
191
A1 A2 A3 A4
190
4-bit-binary-full-adder
C0=1
SN 7483
195
196
reg c vgl
to 15
C4
B1 B2 B3 B4
192a 192b 192c 192d
from 181 of c-register 127

Fig. 15

4 - bit - binary - full - adder SN 7483 truth table									
inputs				outputs					
				when C0 = 0		when C2 = 0	when C0 = 1		when C2 = 1
A1 / A3	B1 / B3	A2 / A4	B2 / B4	S1 / S3	S2 / S4	C2 / C4	S1 / S3	S2 / S4	C2 / C4
0	0	0	0	0	0	0	1	0	0
1	0	0	0	1	0	0	0	1	0
0	1	0	0	1	0	0	0	1	0
1	1	0	0	0	1	0	1	1	0
0	0	1	0	0	1	0	1	1	0
1	0	1	0	1	1	0	0	0	1
0	1	1	0	1	1	0	0	0	1
1	1	1	0	0	0	1	1	0	1
0	0	0	1	0	1	0	1	1	0
1	0	0	1	1	1	0	0	0	1
0	1	0	1	1	1	0	0	0	1
1	1	0	1	0	0	1	1	0	1
0	0	1	1	0	0	1	1	0	1
1	0	1	1	1	0	1	0	1	1
0	1	1	1	1	0	1	0	1	1
1	1	1	1	0	1	1	1	1	1

Note 1: Input conditions at A1, A2, B1, B2 and C0 are used to determine outputs S1 and S2, and the value of the internal carry C2.
The values at C2, A3, B3, A4 and B4 are then used to determine outputs S3, S4 and C4.

Fig. 16

truth-table for SN 7445
BCD-to-decimal-decoder

inputs				outputs									
D	C	B	A	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1	1	1
0	1	1	0	1	1	1	1	1	1	0	1	1	1
0	1	1	1	1	1	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0
1	0	1	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0	1	1	1	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1

truth-table for SN 7473
J-K-master-slave-flip-flop
(each flip-flop)

tn		tn + 1	
J	K	Q	$\overline{Q}$
0	1	0	1
1	0	1	0

tn = Bit time before clock pulse

tn + 1 = Bit time after clock pulse

Fig. 29

Low input to clear sets Q to logical 0.
Clear is independent of clock.

Fig. 17

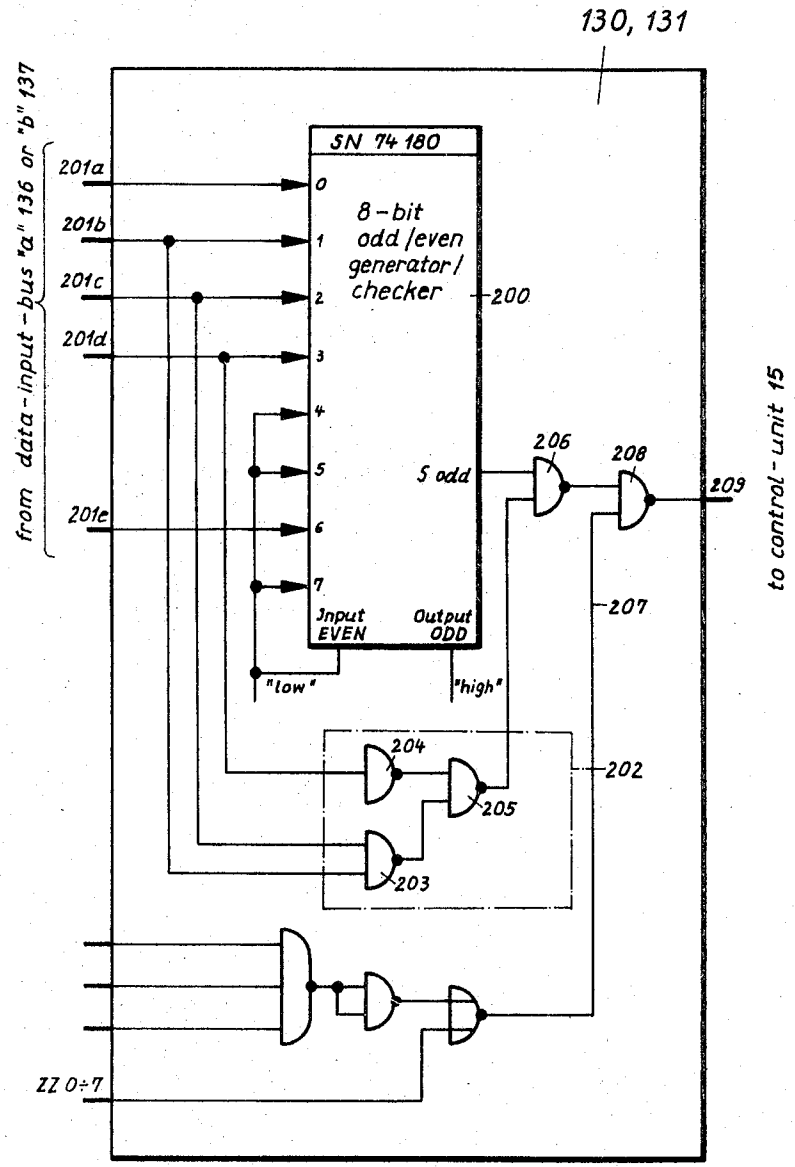

130, 131
SN 74 180
8-bit odd/even generator/ checker
200
from data-input-bus "a" 136 or "b" 137
201a
201b
201c
201d
201e
0
1
2
3
4
5
6
7
S odd
Input EVEN
Output ODD
"low"
"high"
206
208
209
to control-unit 15
207
202
204
205
203
ZZ 0÷7

Fig. 18

200

Truthtable for 8-bit odd/even checker SN 74 180			
Inputs			Outputs
S of 1st. AT 0 thru 7	EVEN	ODD	S ODD
EVEN	0	1	1
ODD	0	1	0

Fig. 20

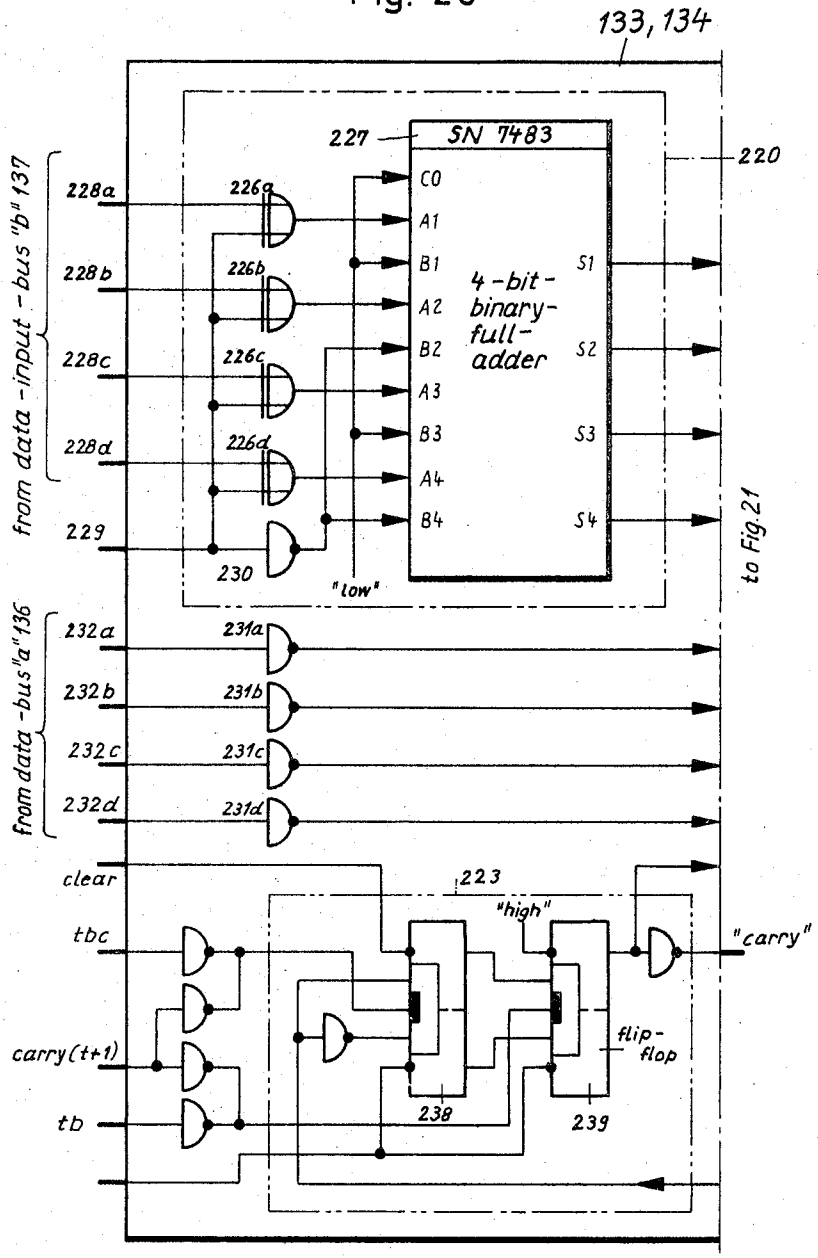

133, 134
227
SN 7483
220
CO
A1
B1
A2
B2
A3
B3
A4
B4
S1
S2
S3
S4
4-bit-binary-full-adder
226a
226b
226c
226d
228a
228b
228c
228d
229
230
"low"
from data-input-bus "b" 137
to Fig. 21
232a
232b
232c
232d
231a
231b
231c
231d
from data-bus "a" 136
clear
223
"high"
tbc
carry(t+1)
tb
"carry"
flip-flop
238
239

Fig. 21

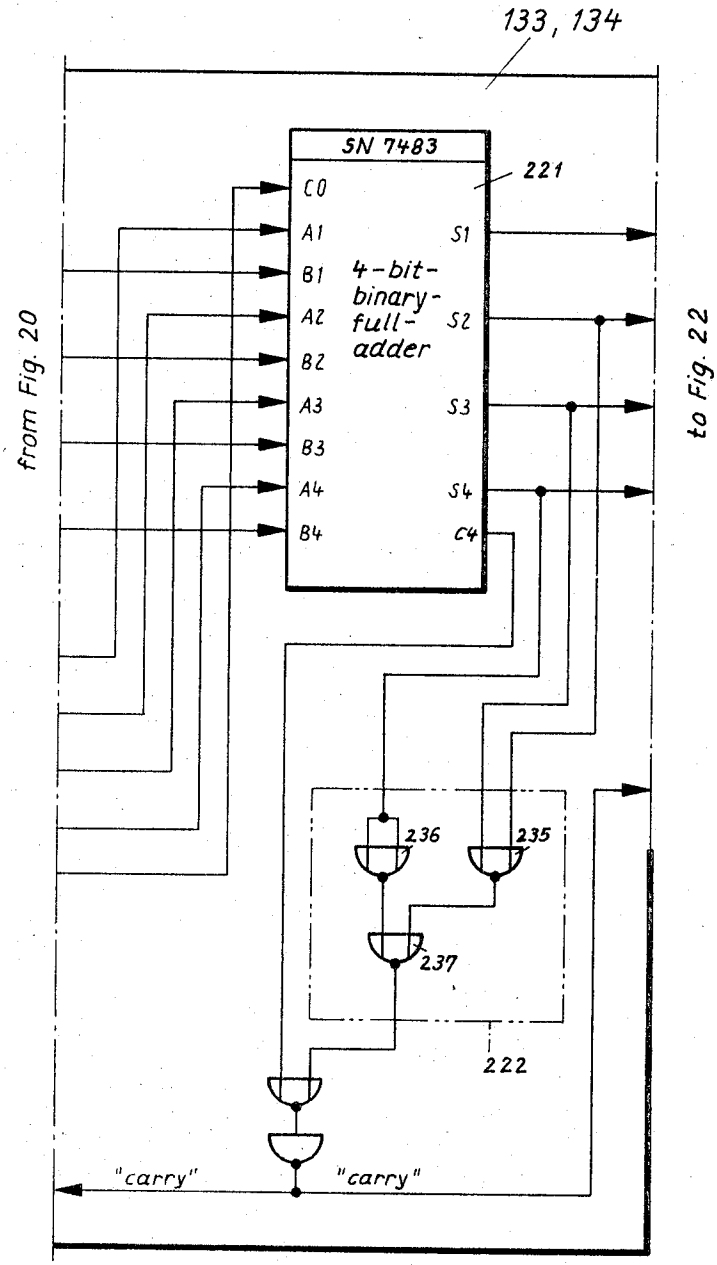

133, 134
SN 7483
221
C0
A1
B1
A2
B2
A3
B3
A4
B4
4-bit-binary-full-adder
S1
S2
S3
S4
C4
from Fig. 20
to Fig. 22
236
235
237
222
"carry"
"carry"

Fig. 22

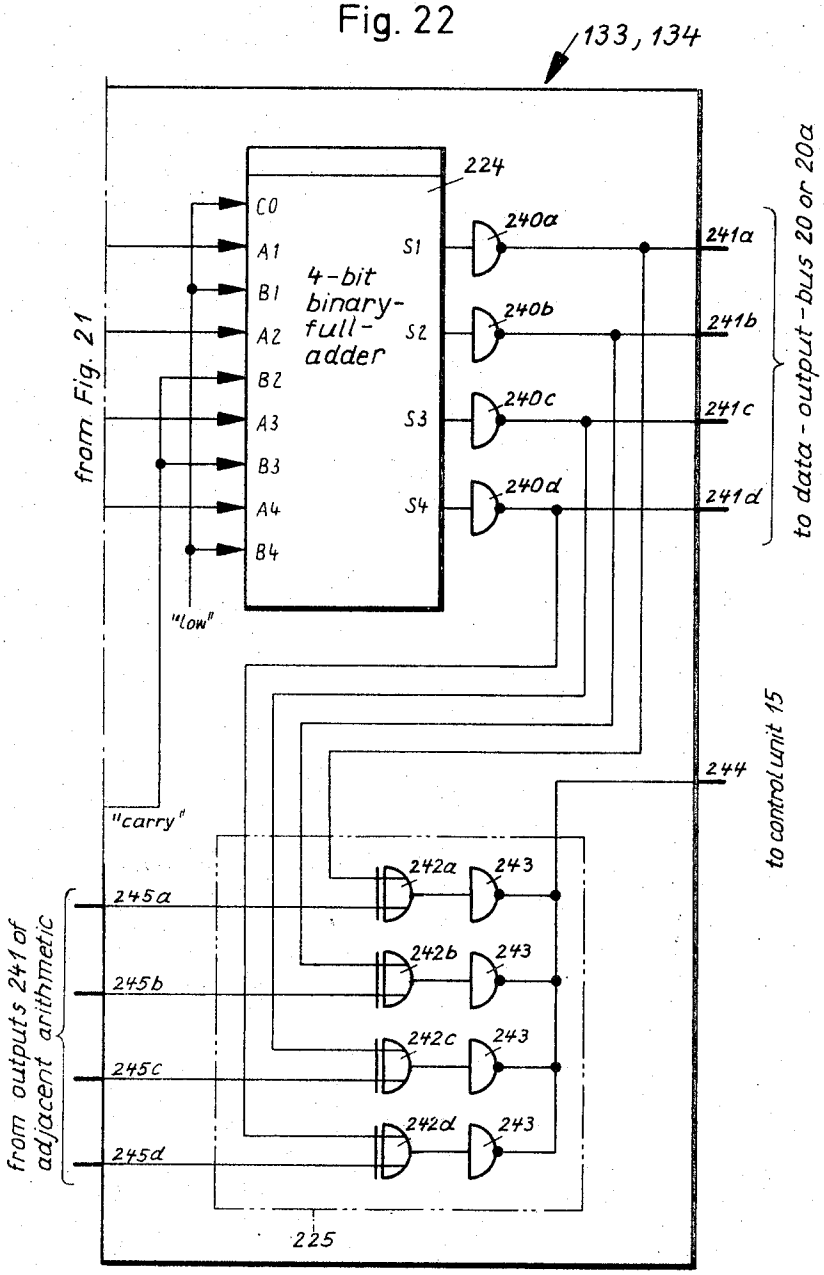

133, 134
224
C0
A1
B1
A2
B2
A3
B3
A4
B4
S1
S2
S3
S4
4-bit binary-full-adder
240a
240b
240c
240d
241a
241b
241c
241d
to data-output-bus 20 or 20a
from Fig. 21
"low"
"carry"
244
to control unit 15
242a
242b
242c
242d
243
245a
245b
245c
245d
from outputs 241 of adjacent arithmetic
225

Fig. 23

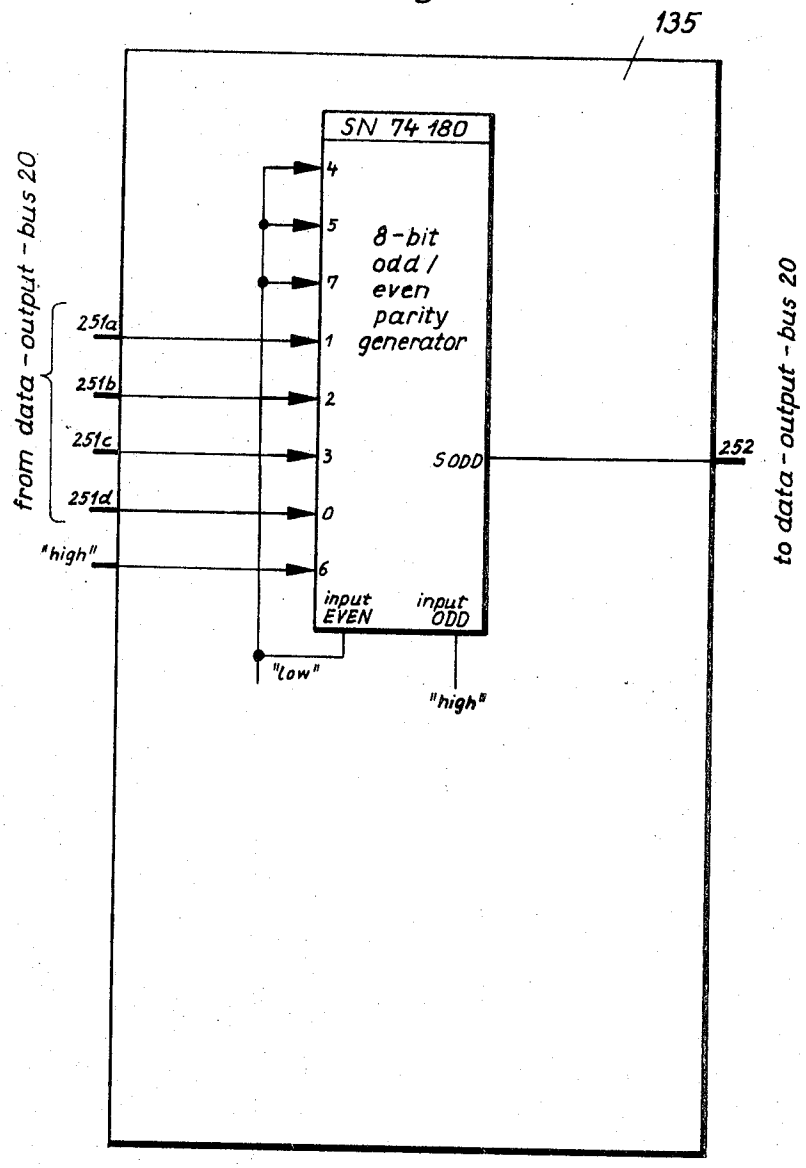

135
SN 74 180
8-bit odd / even parity generator
4
5
7
1
2
3
0
6
SODD
input EVEN
input ODD
"low"
"high"
from data - output - bus 20
251a
251b
251c
251d
"high"
252
to data - output - bus 20

Fig. 24

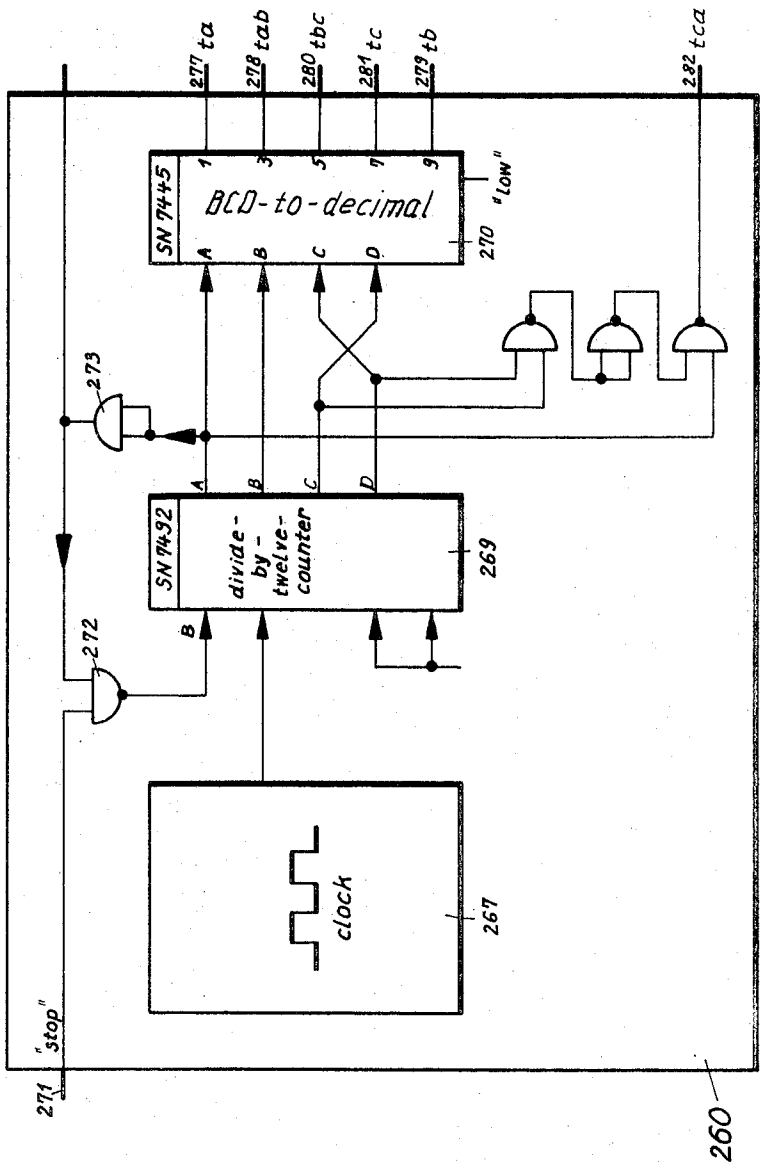

277 ta
278 tab
280 tbc
281 tc
279 tb
282 tca
SN 7445
BCD-to-decimal
270 "low"
A
B
C
D
273
SN 7492
divide-by-twelve-counter
269
272
clock
267
"stop"
271
260

Fig. 25

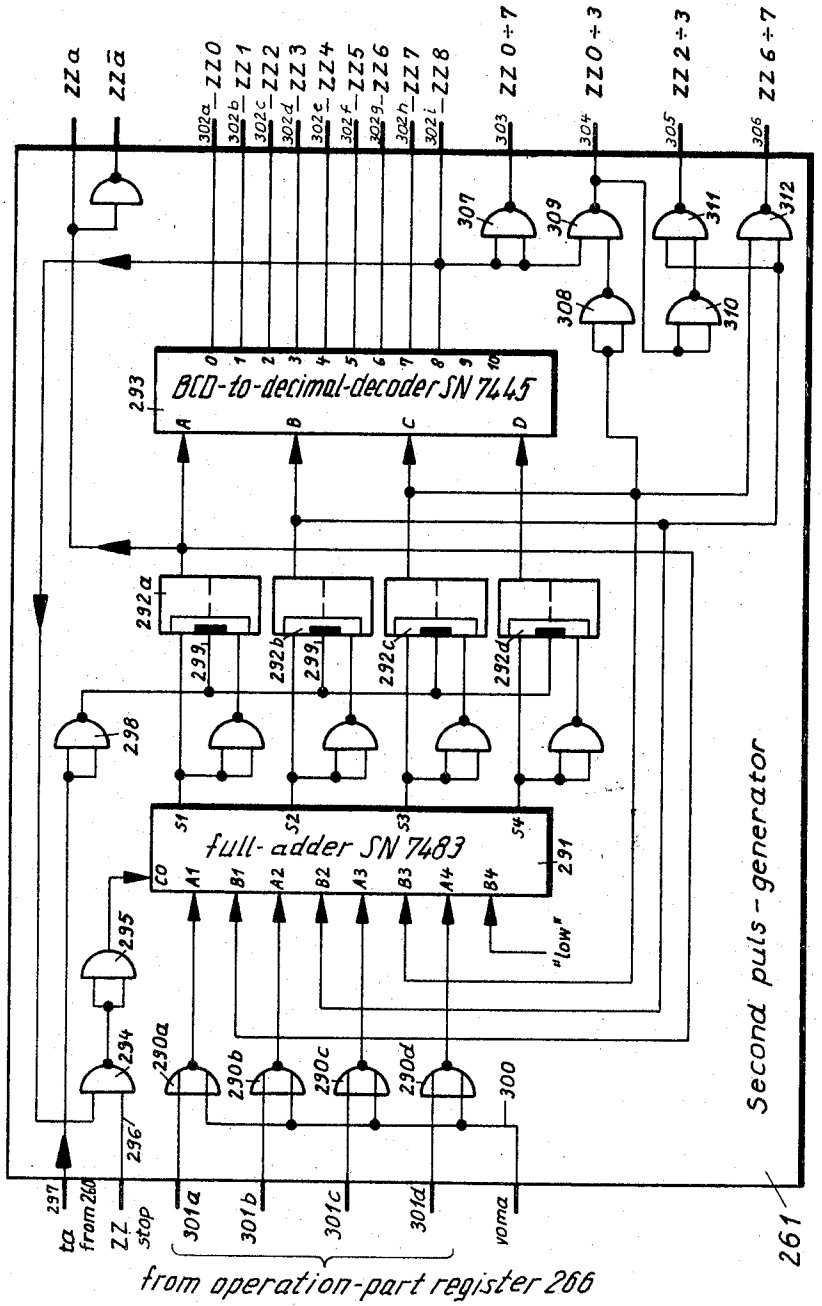

ZZa
ZZ ā
302a ZZ0
302b ZZ1
302c ZZ2
302d ZZ3
302e ZZ4
302f ZZ5
302g ZZ6
302h ZZ7
302i ZZ8
303 ZZ 0÷7
304 ZZ0÷3
305 ZZ 2÷3
306 ZZ 6÷7
307
308
309
310
311
312
293
BCD-to-decimal-decoder SN 7445
A
B
C
D
292a
292b
292c
292d
298
299
S1
S2
S3
S4
full-adder SN 7483
CO
A1
B1
A2
B2
A3
B3
A4
B4
291
"low"
295
234
290a
290b
290c
290d
300
237
from 260
296
ZZ stop
301a
301b
301c
301d
voma
Second puls-generator
261
from operation-part register 266

Fig. 26

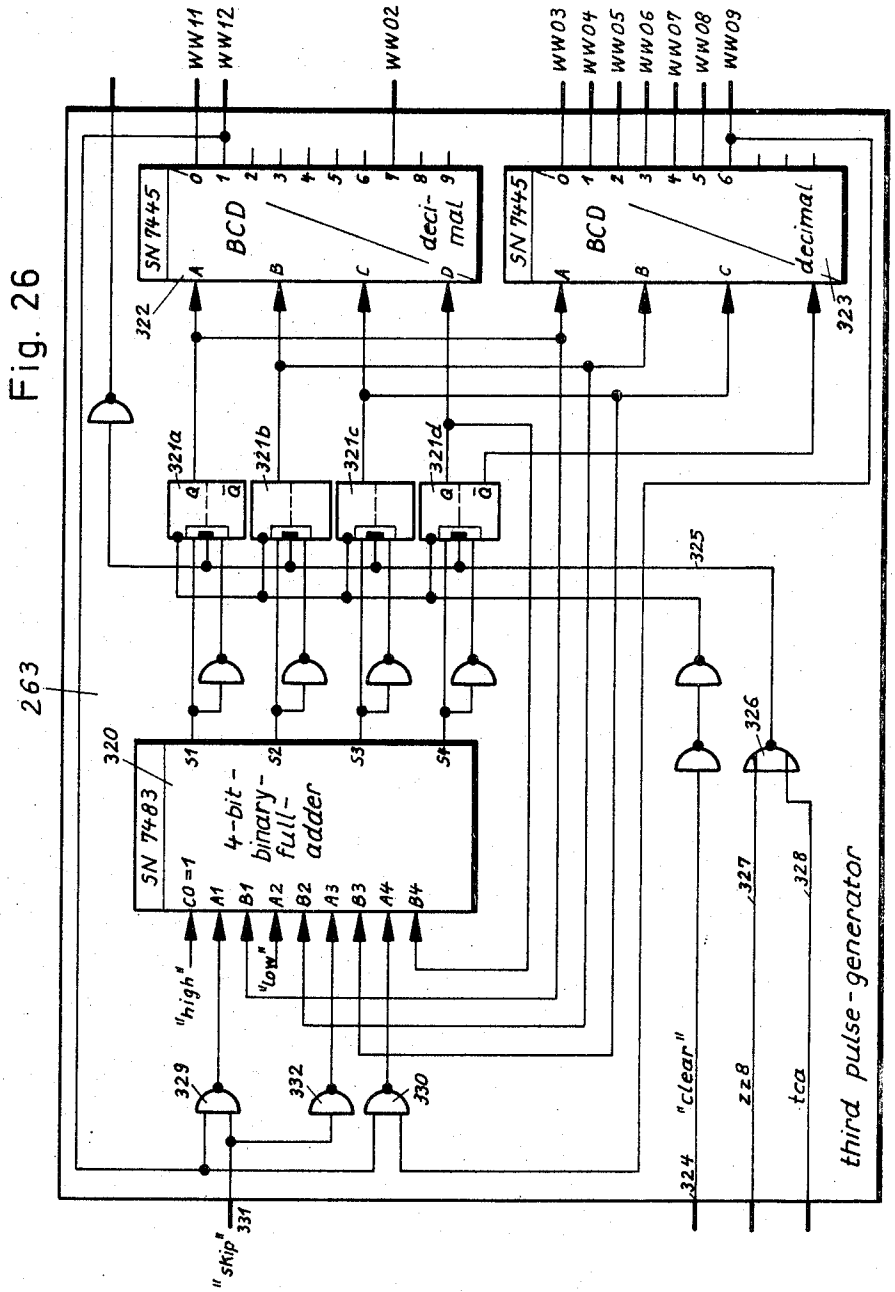

263
WW11
WW12
WW02
WW03
WW04
WW05
WW06
WW07
WW08
WW09
SN 7445
BCD
deci-mal
SN 7445
BCD
decimal
322
323
321a
321b
321c
321d
325
326
320
SN 7483
4-bit-binary-full-adder
CO=1
A1
B1
A2
B2
A3
B3
A4
B4
S1
S2
S3
S4
"high"
"low"
329
332
330
331
"skip"
324
"clear"
327
zz8
328
tca
third pulse-generator

Fig. 27

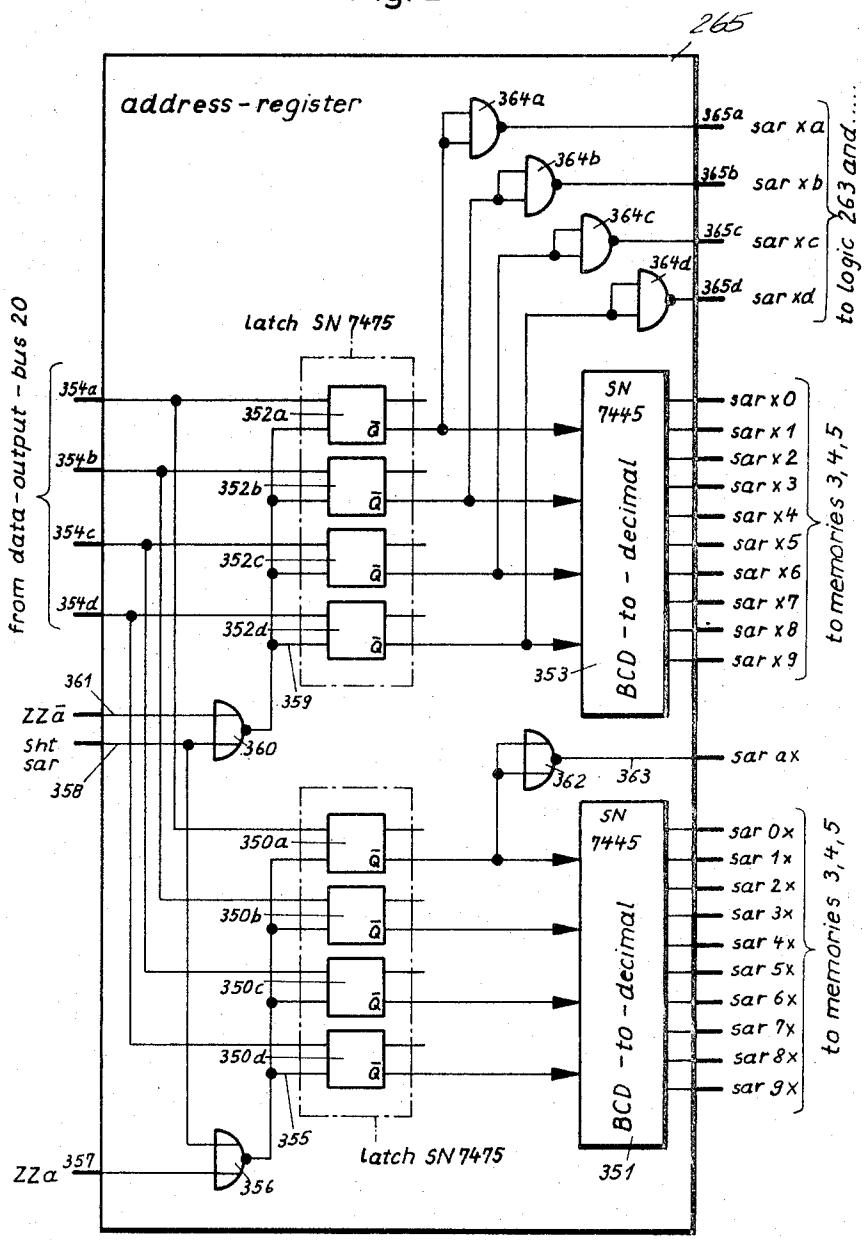
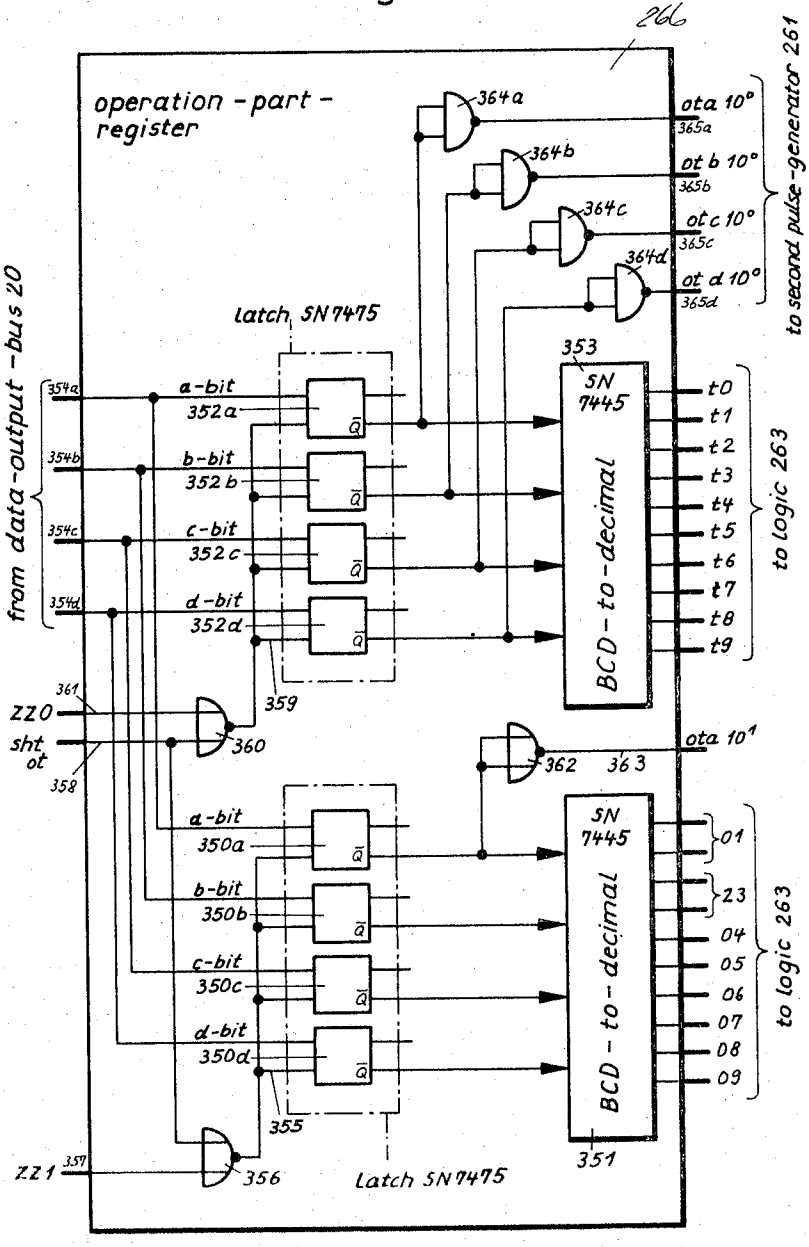

265
address-register
364a
364b
364c
364d
365a sar xa
365b sar xb
365c sar xc
365d sar xd
to logic 263 and.....
latch SN 7475
354a
354b
354c
354d
352a
352b
352c
352d
SN 7445
BCD-to-decimal
353
sar x0
sar x1
sar x2
sar x3
sar x4
sar x5
sar x6
sar x7
sar x8
sar x9
to memories 3,4,5
from data-output-bus 20
361
ZZā
Sht sar
358
359
360
362
363
sar ax
350a
350b
350c
350d
355
latch SN 7475
ZZa
357
356
351
sar 0x
sar 1x
sar 2x
sar 3x
sar 4x
sar 5x
sar 6x
sar 7x
sar 8x
sar 9x
to memories 3,4,5

Fig. 28


266
operation - part - register
364a
364b
364c
364d
ota 10°
365a
ot b 10°
365b
ot c 10°
365c
ot d 10°
365d
to second pulse-generator 261
latch SN7475
353
SN 7445
BCD - to - decimal
t0
t1
t2
t3
t4
t5
t6
t7
t8
t9
to logic 263
from data-output -bus 20
354a
354b
354c
354d
a-bit
352a
b-bit
352b
c-bit
352c
d-bit
352d
359
361
ZZ0
sht ot
358
360
362
363
ota 10¹
a-bit
350a
b-bit
350b
c-bit
350c
d-bit
350d
355
SN 7445
BCD - to - decimal
01
23
04
05
06
07
08
09
to logic 263
351
ZZ1
357
356
Latch SN7475

WEIGHING AND PRICE CALCULATING DEVICE

The present application is a continuation-in-part of application Ser. No. 87,570 filed 11/6/1970, now abandoned.

BACKGROUND OF THE INVENTION

This invention relates to a weighing and price calculating device with an arrangement for printing the signalled weight values and the prices calculated therefrom on tickets, labels or the like which are to accompany the weighed goods.

Weighing and price calculating devices are known having printing arrangements which are operated thereby. An assembly of this kind is described in detail in German Offenlegungeschrift No. 1,474,631. This weighing and price calculating device has a numeral selector switch enabling a three-place decimal number to be preselected. When the type of goods which are to be weighed changes a number of times each day the price per weight unit has to be varied at the selector switch with a corresponding frequency, which takes up a lot of time and also involves the danger of a faulty adjustment and consequently wrongly calculated end prices. Frequently it is desirable that the price per weight unit shall be selectable to four decimal places. In this case the changeover takes even more time and the danger of a faulty setting of the selector switch is much greater.

To avoid these drawbacks, the invention is concerned with a weighing and price calculating device having a calculator and, connected to the latter, a data distributor provided with a printer for printing signalled weights, and prices calculated therefrom, on tickets or the like which are to accompany the weighed goods, and is characterized by a memory which contains addresses and in which at least two selectable numerical values can be stored, and by an address selector switch for selectably connecting the addresses to the calculator.

An advantageous embodiment of a weighing and price calculating device according to the invention is, in accordance with a further feature of the invention, characterized by the fact that the memory containing the addresses comprises at least two multi-position numeral selector switches of a type known per se.

In accordance with a further feature of the invention a device of known form to guard against unintentional adjustment is provided at the numeral selector switches.

A further advantageous embodiment of the device according to the invention is characterized by the fact that the memory containing the addresses comprises a punched card or punched strip-reader.

Advantageously in accordance with the invention, the data distributor has associated therewith a printer, known per se, for printing out the names of the goods and adapted for the reception of interchangeable or replaceable printing plates each carrying the name of a commodity, and the printer has arranged thereon a code reader for detecting markings provided on the printing plate and controlling the selection of the address to be chosen from the memory; further in accordance with another feature the code reader has associated therewith a code-actuated selector switch (known per se), and the code reader directly controls the selection of addresses from the memory through this code-actuated selector switch.

The memory containing the addresses may, in accordance with yet another feature of the invention, in addition to its use for storing values of price per weight units, also be used for storing and feeding tare values for packages, and other pre-selectable numerical values.

The weighing and price calculating device according to the invention above all has the advantage that the changeover from one type of commodity to be weighed to another such commodity can be performed in a matter of seconds by simple interchange or replacement of printing plates, while at the same time a faulty resetting of the prevailing price per weight unit value, or some other selectable numerical value, is inhibited.

An advantageous embodiment of a price marking weighing device in accordance with the invention is hereinafter described with reference to the accompanying drawings which include several block wiring diagrams.

BRIEF DESCRIPTION OF THE FIGURES

FIG. **1** is a schematic wiring diagram showing the entire weighing and calculating system with several of the circuit components shown in block form.

FIG. **2** is a schematic diagram of the calculator shown in FIG. **1.** This view shows the calculator circuit in greater detail.

FIG. **3** is a diagram showing the control unit in greater detail, including some of the wave forms generated by the three pulse generators.

FIGS. **4** and **5** are schematic diagrams of connections showing the components of the numerical memory which receive and record the weight values from the weight signalling unit in FIG. **1.** These figures should be joined, FIG. **4** above FIG. **5,** to show the complete circuit.

FIG. **6** is a wiring diagram in greater detail of one of the multiposition numeral selector switches shown in block form in FIG. **4.**

FIG. **7** is a diagram of a code reader **17** and a printer **16,** also shown in block form in FIG. **1.**

FIG. **8** is a diagram and a truth table of an exclusive OR gate indicating the action of this solid state component.

FIG. **9** is a wiring diagram, partly in block form, of the memory units for storing instructions and constants to be used by the calculator.

FIG. **10** is a wiring diagram of the bar register shown in block form in FIG. **2.**

FIG. **11** is a block diagram of a flip-flop used in the shift register shown in FIGS. **10** and **12.** FIG. **11** also shows the truth table of this component, indicating its operation.

FIG. **12** is a wiring diagram of the B register **128,** shown in block form in FIG. **2.**

FIG. **13** is a wiring diagram of the A register **126** and the C register **127.** These components are shown in block form in FIG. **2.**

FIG. **14** is a wiring diagram of the comparator unit **132,** shown in block form in FIG. **2.** This unit includes a binary counter and a four-bit binary full adder.

FIG. **15** is a truth table of the four-bit full adder, defining the output voltages in terms of the input voltages.

FIG. 16 is a truth table for the binary to decimal decoder, also shown in block form in FIG. 2.

FIG. 17 is a wiring diagram of the parity checkers 130 and 131 shown in FIG. 2.

FIG. 18 is a truth table for the odd/even generator checker shown in block form in FIG. 17.

FIG. 19 (on sheet 7) is a detailed wiring diagram of the comparator 132 shown in block form in FIG. 2.

FIGS. 20, 21 and 22 are wiring diagrams of either of the arithmetic units 133,134 shown in block form in FIG. 2. The three figures should be joined as indicated to show the complete circuit.

FIG. 23 is a block diagram showing the connections to the 8-bit odd/even parity generator 135 shown in FIG. 1.

FIG. 24 is a wiring diagram of the first pulse generator 260 shown in FIG. 3.

FIG. 25 is a detailed wiring diagram of the second pulse generator shown in block form in FIG. 3.

FIG. 26 is a schematic diagram of connections of the logic circuit shown in block form in FIG. 3.

FIG. 27 is a detailed wiring diagram of the address register 265 shown in block form in FIG. 3.

FIG. 28 is a similar diagram of the operation - part register 266 shown in block form in FIG. 3.

FIG. 29 (on sheet 15) is a truth table of the slave flip-flop SN7473 shown in block form in FIGS. 27 and 28.

DESCRIPTION OF THE PREFERRED EMBODIMENT

The invention will first be described generally, referring to FIG. 1.

The weighing and price calculating device which is illustrated by this figure comprises a weigher 1 of known form incorporating damping means and a weight signalling unit 2, again of a known type, set to transmit digital weight readings; a memory 3 for the weight readings connected directly to the weight signalling unit 2; a memory 5 with addresses into which can be fed freely selectable numerical values, for example for price per unit weight, tare weight and permissible weight limits; a memory 6 for receiving instruction elements and numerical values which can be presecribed in advance; a calculator 13 which is connected to the addresses of the memories 3 and 6, through conductors 8, 9, 11, 12 and has or consists of a calculating unit 13 and code producers (not shown) of a form known per se, together with scanning devices of known type, such as comparators and code readers; an output unit 14 arranged next to the calculator 13; a control unit 15 of a form known per se, and including a clock generator (not shown) of known type, for governing the addresses of the memories 3 and 6, for reading memory cells thereof and, where appropriate, for extinguishing memory cells, controlling the calculator 13 and the output 14, and for demobilizing these two latter in the event that faulty indications on the control mechanism are reported by the scanning devices; and a printer 16 for printing out names of goods and having a code reader 17 arranged thereon.

The memory 3 which can receive freely preselectable numerical values preferably comprises a plurality of groups of multi-position numeral selector switches 18, of a type which is freely available in commerce, having binary outputs, so that at least two price per weight units can be set independently of one another, and in addition numeral values for tare, permissible weight limits, limits for end prices, and so on can be set if required.

The memory 5 which is intended to be fed with freely selectable numerical values may however be a known form of reader for freely-markable recording carriers, for example a punched card or a punched strip-reader.

The calculator 13 is preferably set up for calculating with decimally encoded (BCD) binary figures and is connected to the control unit 15 through data conductors 40, 41 such that calculating steps can be performed in a predetermined way with the aid of a program fed into memory 6.

The output unit 14 has a data printer 19, of a form known per se, and it is connected to calculator 13 by a conductor 20.

In addition, the data distributor 14 may be provided with a visual indicating appliance (not shown) and recording appliances of known form.

The printer 16 for printing the names of goods, and words which are independent of the weight, operates through the medium of non-varying printing plates which are not shown in the accompanying figures. These plates may either be individually replaceable or can be disposed on a drum to enable them to be changed. The code reader 17 is arranged on the printer 16 to detect markings from a printing plate disposed in the printing position and, through a conductor 21, operates an address selector switch of known form (not shown) in the memory 5 for price per weight unit etc. When the number of addresses in the memory 5 is small the markings on the printing plates are applied in accordance with a required one-from-n code, and the code reader 17 directly controls the address selector switch. If the number of addresses in the memory 5 has to be large because it is required to signal the price for goods of a very large number of commodities, a binary code is then used for marking the printing plate and a code converter of known form, but not here shown, is arranged after the code reader 17 and in turn controls the addresses in the memory 5.

A photoresistor unit 22 is provided on the weigher 1 and this is operated by packages 24 or objects which are fed individually over the weigher 1 by means of a conveyor belt 23 when it can be anticipated that the weigher 1 has come to rest in a balance condition corresponding to the load of the individual package concerned.

Operation of light unit 22, through a conductor 25, starts the start control unit 400 which then transmits control instructions to the control unit 15, the calculator 13, and memories 3, 5 and 6 through conductors 26 and 366.

The first control instructions have the effect that predetermined numerical values are read from the memory 6 and transmitted to the calculator 13 which then performs a control multiplication with these values. The control multiplication is simultaneously performed in the two arithmetic units of the calculator 13 and the results are first examined for equality. When the results are the same, these are then compared with the number which has been fed into memory 6 and represents the target result.

The weight signalling unit 2 is then operated through a control conductor 38 and this action sends a weight

indication into the weight reading memory **3**, through a data conductor **39**.

Thereafter the calculator **13** is operated in such a way that the weight values are sent to the weight reading memory **3**, and is then provided with a code bit, which is transferred into a shift register (not shown) of known form in the calculator **13**.

A further operating step is implemented by the fact that a further weight, which is given by the weight signalling unit **2** shortly after the call for the first, is requested from the weight reading memory **3** and likewise transmitted into a further shift register of calculator **13**. The first and second weight values are compared. If there is a difference exceeding a preselected and stored amount, the calculator **13** transmits a signal to the control unit **15** through a conductor **33**, and this stops calculator **13**. When the comparison is satisfactory the second weight reading is preferably retained in calculator **13**. Further operating commands contained in memory **6** then have the effect that the calculator compares the weight value with numerical values which can be determined from the memory **6** or **5**. Thus the numerical values represent a limitation of the technically available weight range of the weigher **1** so that only packages **24** are approved which have a weight which does not exceed a pre-selected figure. Thus a maximum permissible end price can be indirectly preselected even when account is taken of the price per weight.

If the weight is outside the presecribed limits the calculator **13** is stopped and the package **24** concerned is rejected.

When the check finds that the weight reading lies within the prescribed limits, a predetermined tare value for packaging or the like may, if required, be subtracted from the weight reading so that only the numerical value corresponding to the net weight is used in evaluating the end price. Numerical values for the tare value may either be given directly to memory **5** or may be set, as appropriate, on the figure selector switches **18** of memory **5**. If a number of tare values are to be stored, one of them can likewise be chosen by the code reader **17**, in the way provided for in connection with the selection of the price per weight unit. Correspondingly, a number of numerical values for the weight limits could be pre-selected and then likewise be selected by the code reader **17**. Further, it is also feasible to transmit the tare for each package before the latter is filled, to feed it into memory **7** and then to call it up as is normal in filling appliances such as that of Swiss Pat. Specification No. 383,866.

This weight reading of weigher **1**, if corrected, is also written into output unit **14** via conductor **20** and is then multiplied by the selected price per weight unit selected from code reader **17**.

The multiplication product is rounded up or down in chosen fashion by programmed further calculating steps. Thus, for example, decimal values of cents may be rounded and printed out in quantities of whole number multiples of the numeral 0.5. If appropriate, other program steps can be provided for by means of which, from the price calculated in a currency which does not conform to the decimal system in the smallest unit, there is subtracted in successive steps that number which corresponds to the increase from the smaller unit to the next larger unit. This subtraction is repeated until a remainder is left which is smaller than the increase from the smaller to the larger unit. The number of possible steps during the subtraction and the remainder which is left then constitutes the indicated price.

The calculated end price is then likewise transferred to the data distributor **14** and subsequently printed out, along with the previously quoted weight indication and the price per weight unit, on a ticket or the like (not shown) by means of the printer **19**. The data distributor **14** is connected through conductors **40, 41** with the control unit **15** which controls and operates this distributor.

If required, moreover, weights and end prices of a number of weighed packages **24** can be sent through conductors **20, 35** into memory **7** for summation purposes so that the total weight and total price of a collection of packages can be calculated and likewise printed on tickets. There is also the further possibility that provision may be made in memory **7** of addresses for customers accounts and for the storage of the total weights and total prices of a collection of packages, in accordance with the distribution among customers, for the purpose of automatic production of delivery notes and bills.

As has already been said earlier in this specification, the weighing and price calculating device in accordance with this invention has the advantage that when changing from one type of commodity to another only the printing plate with the name of the commodity needs normally to be changed. Should the price per weight unit vary in one particular type of commodity this new price can readily be set on the numeral selector switches **18** of memory **5**.

The fact that the control unit **15** which is used to control the calculator **13** can be storage programmed, enables a comparatively extensive distribution program to be carried out. If the program has to be adapted to a changed business situation it would be sufficient to replace the memory **5** with its fixed program by a similar memory provided with a different program. A reconstruction of the control unit **15** and the calculator **13**, or the installation of additional switching circuits, are thus not necessary. Moreover the use of the control unit **15** which can be storage programmed has the advantage that it is not necessary to make any mechanical modifications to the printer **19** if the rounding up or down of calculated prices has to be varied.

DETAILED DESCRIPTION

FIG. **2** shows the calculator **13** in more detailed form, dividing this circuit into eleven sub-circuits, still shown in block form. FIG. **3** shows the control unit **15** in more detailed form, dividing this circuit into seven subcircuits, also shown in block form. The other figures provide detailed circuits of the components shown in FIGS. **2** and **3** and give additional information regarding its operation.

Referring now to FIGS. **2** through **7** and **9**, the calculator **13** (FIG. **2**) comprises three registers, A, B, and C, **126, 128,** and **127,** a bar register **125,** and a comparator and step counter **129.** In addition, there is a first arithmetic unit **133,** a second arithmetic unit **134,** a parity bit generator **135,** two parity checkers **130** and **131** and a comparator **132** which compares the values in the two arithmetic units. FIG. **3** shows the main components of the control unit **15** which comprises a logic circuit **263,** a comparator and step counter **264,** an address register **265** and an operation-part register **266.**

This circuit also includes a first pulse generator **260,** a second pulse generator **261,** and a third pulse generator **262.**

FIGS. **4** and **5,** when joined together show the wiring details of the "memory for selectable numerical values" storage unit **5** shown in block in FIG. **1.** This circuit contains a plurality of numeral selector switches **18,** 12 of these being shown in FIG. **1** but only four indicated in FIG. **4.** The details of these switches are shown in FIG. **6.** FIGS. **4** and **5** also show the input gates which control and direct the operating pulses to these storage units.

The code reader **17** and printer **16** are shown in detail in FIG. **7** and include the usual switches and type bars common to all printing arrays. The three registers A, B, and C (FIG. **2**) **126, 127** and **128** are used for multiplication calculations. The wiring details of these registers are shown in FIG. **13.** Each register employs five 8-bit shift registers of the type SN 7491N which are available from the Texas Instrument Corp. The inputs for these registers come from the start control circuit **400** over conductor **399** and from the conductors in bus **20.** The registers are arranged with input and output gates so that the shift operations are controlled by clock pulses from generators **260, 261** and **262.**

The unit **129** for counting and comparing has a ripple-through-counter, not shown, similar to the type SN 7493 N and a 4-bit-full-adder of the type SN 7483 N, which are connected by means of gates, to each other in such a manner that pulses being fed into the multiplication counter are compared with a BCD-number being fed into the second input of the full adder of the comparator **134.**

The No. **9** complementer **133A** and **133B** are composed of exclusive OR-gates and full-adders of the type SN 7450 and SN 7483.

Each of the arithmetic units **133, 134** consists of a first 4-bit-full-adder **227** (FIG. **20**) of the type SN 7483 of the firm TEXAS INSTRUMENTS, by means of which each 4 bits of two numbers are to be added, and of a second 4-bit-full-adder **221** (FIG. **21**), which serves to transform calculating results from the first full-adder, which are larger than the decimal value **9,** by addition of binary 5 in BCD and a bit for transmitting - carrying into a decimal position of higher order. Basically, ony one arithmetic-unit is needed for adding two numbers. If, however, there are provided two arithmetic-units, it is possible to perform the same addition by both at the same time. The addition results being calculated independently from each other are thereafter compared by means of the comparator **132** and only then, if the latter are equal, one can be sure that it was calculated correctly. To calculate two times with only one arithmetic unit would lead to the disadvantage that the same error can arise and would therefore, remain unknown.

The comparator **132** is a so-called exclusive OR-gate, for instance of the type SN 7450 and is shown in detail in FIG. **19** on sheet **7.**

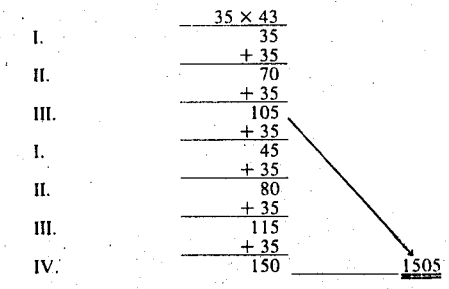
An addition of two numbers having a plurality of denominational orders is performed in such a manner that these numbers, beginning with the lowest order number, are fed into the registers A and B **126** and **128** respectively and are set so that zeros are entered before the digits in the highest order. The numbers with the lowest order appear thereby on the outputs of the registers A, B. The numbers are sent over conductor **12** through a switch **61** into the register B and additionally through the arithmetic unit **133** into register A. Thereafter, the numbers with the lowest value (least significant digit) are shifted out of the registers A and B and are fed through bus **20** to both the arithmetic-units **133, 134.**

The shift-out is achieved by the fact that by means of a clock-pulse the contents of the registers are shifted by one order for each pulse. Numbers being shifted out from the register B are sent back to its input and stored in the first register stage. If this operation is repeated as often as there are register-stages (in the present example **8** stages) the number is present again in the register B in such a manner as it was present before the beginning of the calculating operation.

Within the arithmetic-units **133, 134** the fed numbers are added and, if the result is present as dual-number, which is larger than decimal **9,** it is then transformed into BCD and a transmission-carry. The results in BCD-form are fed to the comparator **60** and compared bit by bit. Parallel to this, from the arithmetic unit **58** a result is fed to the input of the register A and is fixed in its first storage stage. If there is realized an equality by the comparator **132,** the calculator **13** is shut off. The addition is continued by shifting the numbers in the registers A and B by one order, whereby both numbers with the next higher order arrive in the arithmetic units **133, 134** and are added by means of the first full-adder.

The results of the both arithmetics are again compared to each other by comparator **132.** A result is fed again to the register A. After a number of addition steps corresponding to the number of stages of the registers, the register A contains the addition result with the lowest order on the output. The result can thereafter be transmitted to the output unit **14** in that the result is shifted out order-by-order from the register A and is fed without alteration through the arithmetic unit **133** to the conductor **137.**

Multiplications are to be achieved as a sequence of additions according to a preselected scheme. For instance the result of 43 × 35 can be achieved as follows:



First the number **35** is brought into the register B and the number **43** is brought into the register A and are supplemented by zeros before the numbers with the highest order in such a way that all storage orders of the registers A and B are filled and the numbers with the lowest order exist at the output of the registers A, B. Thereafter, the number **43** is shifted into the end of the register C, whereby the register A is emptied. The number **53** of the multiplication counter **129** and comparator **132** is then set to zero. Thereafter, the real multiplication operation begins, which is divided into single addition steps.

It is next determined, whether the last number in the register C corresponds to the counter position "**0**". If this should not be the case, the contents of the registers B and A are fed in a manner described above to the arithmetic-units **133, 134** and the result is stored in register A. Register A also contains the number **35** at this time. This addition operation is modified by a pulse applied to the multiplication counter **129** and sets this to "**1**". Now a comparison is made to determine whether the indication of the counter **129** is in conformity with the last number in register C. In the present case the numbers are unequal and the contents of the registers B and A are added again and stored in the register A. Register A now indicates **70.** During the addition operation the counter **129** was switched over again by one step and indicates now "**2**". A comparison shows still no conformity between the counter-indication and the last number in register C. The addition operation is thereafter repeated. Register A now contains the number "**105**" and the counter-indication is changed to "**3**". By comparing there is shown equality. Thereafter the achieved partial result is shifted in the register A in the direction to the output by one order so that the number "**5**" with the lowest order-value is shifted over into the input of the register C as the highest value-order thereof. At the same time, the original content of the register C is shifted by one order so that the last number "**3**" is omitted, whereby it is replaced by the number "**4**" of the number "**43**". Thereafter the counter **129** is set to "zero" and the contents of the registers B and A are added and the result is stored in register A. Register A now contains the number "**45**". To this number "**45**" the "**35**" is added until the number of the addition operations corresponds to that of the last number in the register C, in this case "**4**". The register A now contains the number "**150**". Thereafter the contents of the registers A and C are shifted again by one order. From the register A is transmitted by this means the "**0**" into the highest order position of the register C and in its highest but one position is now the number "**5**".

The number "**4**" on the end of the register C is shifted out and is omitted. If a comparation is performed with the counter **129** being set again to "zero" equality is determined. No addition is performed. Instead of this, the registers A and C are shifted again.

By this the "**5**" goes from register A to register C. Register C now contains the number "**505**". The last filling "**0**" has been shifted out and replaced by the following digit. By repeating the last described step the "**1**" travels out of the register A into the register C so that this contains the numbers "**1**"; "**5**"; "**0**" and "**5**". The number of the shifting steps is limited by cycles counter **264** (FIG. **3**) in the control unit **15,** whereby this counter allows only so much shifting steps as the register A has positions. After having performed this number of steps, the "**5**" is then shifted at first into the register C in the lowest order position of this register and the multiplication operation is ended. The result "**1505**" is contained with the correct orders in the register C.

Subtractions can be done by the combination of a complementing of the subtrahend and an addition.

For example, 348 − 327 can be calculated in the decimal system as follows:

348 − 327 =
= 348 + (999 − 327) + 1 − 1000
+ (348 + 672 + 1) − 1000
= (1021) − 1000
= **21**

The number "**1**" on the above calculation is termed the "fugitive" **1** and is really the difference between the **9** complement and the **10** complement.

To subtract, the number **348** is added to the register B and the number **327** is stored in register B. The last number of register A is then fed directly to the first full adder of the arithmetic units **133, 134** and the last number from the register B is fed to the No. **9** complementer **133**A and the transforming result being achieved by this is fed in BCD-code also to the full-adders. The additions result is fed to the register A, as has been described.

Thereafter, the next calculating operation is performed on the numbers of the next higher order-value and the addition of the carry from the memory being generated from the lower order-value. When all orders of minuend and subtrahend are calculated, the correction-calculation is performed. The correction-calculation comprises an operation in which the number being contained in the register A is fed again to the arithmetic units **133, 134.** The carry being generated during the last addition operation is then added to the number being shifted out of the register A and the new number is fed as the final result to register A.

If the example would read 327 – 348 and the number 327 would be present in register A and the number 348 in register B, there is no carry in the last addition operation. Thereafter the number from register A must be fed through the complementer **133**A and the transformation result must be given without change through the arithmetic units into the register A. This transformation result is the final result. Because there was no carry, the result is negative.

From the above examples of addition, multiplication, and subtraction, it is evident that each kind of calculation is performed in a predetermined sequence and that an operation of the single elements of the computer is required.

In the course of the addition operation it was apparent that the calculation is performed according to a simple scheme: There are effected additional shifts in the registers A and B, after the arithmetic units **133, 134** are operated. Then the comparation of the partial results from the arithmetic units and the shift of the partial results into the register A is performed. These operations are repeated, up to the point the registers A and B have been shifted through a complete cycle.

In the course of the multiplication after each addition, it must be decided, whether there should be another addition or whether the next step is a shift in the registers A and C. To perform these different operations, it is necessary to connect the registers A, B and C through a series of output gates **240, 242** in a different manner with the arithmetic units and to take care that these registers A, B, C and the memories of the arithmetics are controlled in proper timed sequence.

This connection and the necessary control is performed by means of the control-unit **15.** Unit **15** comprises for this purpose a multi-clock-generator **260, 261** and **262** with a plurality of outputs, a digit-counter **264** having a plurality of outputs, and an array for generating logic terms **263.** An instruction register, which consists of a first part **266** for receiving a command and a second part **265** for receiving a command address.

The pulse generators **260, 261** and **262** are connected to the logic array **263.** Also, the instruction register **266** and the operation cycles counter **264** are connected to the array **263.** The array **263** is controlled by commands. Such commands can be shifted from the memories **6** through the conductors **9, 8, 12,** the arithmetic unit **133** and the conductors **20** and **26** into the instruction registers **264.** The commands are stored in predetermined sequence in the memory **6** and consist of an operational part and an address part. The kind and sequence of the commands and the division of them into operational part and address part is given in programs for commercially operating single-address-large-computers, which consist of a central unit which includes a central processing unit and an operational storage-working memory. Thereby, the operational part of unit **263** can operate to shift numbers into or out of registers A, B, C or memories **3, 5** or **6** respectively, or to perform decisions to jump to locations of the programs (jumps instruction), which do not correspond to the sequence in the storage. Such decisions are made if, in the course of a multiplication, either no additions are to be made or if a shift of orders is to be performed.

In the array **133A** there are contained logic circuits such as NAND-gates, NOR-gates and bistable flip-flops, which are sold as "integrated circuits" of the type - series SN **74** by the firm TEXAS INSTRUMENTS. The logic circuits serve for decoding the commands and to make connections to the single computer elements and for switching pulses from the clock-generator **260, 261,** and **262.** The control commands extend through the conductor **26** to calculator **13.** Messages from the calculator **13** are fed through the conductor **33** to the array **263** of the control unit **15.** The computer **13** and the control unit **15** operate after having shifted a first command into the instruction register in such a way that an operational sequence corresponding to the code of the command is started. At the same time, the address part of the command contains information, from which storage address, if a number is to be carried to a circuit or brought from a register. With the execution of a command, the number of the command address being present in the second part of the instruction register **264** is incremented by "**1**". This is performed by the command address in the instruction register **70,** and is shifted through the arithmetic units **133, 134** and thereby this number is incremented by "**1**". This number being incremented by "**1**" constitutes the address of the next command for the case when the sequence of commands being present in the storage **6** and a further step is to be accomplished. In this case the instruction register **129** is called also an instruction counter.

As has been described above, there must occasionally be a change from the sequence of commands being present in the storage **6.** For this case the jump instruction produces a command address, being incremented by "**1**" in the instruction register **264** and is cleared and replaced by a certain instruction, from which the next command is to be gained. This determined command address is contained in the address of the jump instruction. For the reason that the sequence of commands and the insertion of jump instructions is freely selectable, the operation of the computer can be adapted to other problems.

We claim:

1. A weighing and price calculating device comprising a weigher, a calculator connected to said weigher and adapted to receive signals therefrom corresponding to the weights of goods passed to said weigher, a data distributor connected to said calculator, and a printer coupled to said data distributor for printing out signalled weights and prices calculated therefrom on to labels to accompany the weighed goods; the improvement which includes a first memory connected to the weigher, for receiving a series of digits representing the weight of the goods on the weigher; a second memory in which is stored a series of digits representing the price per unit weight of the material being weighed; a sensing device positioned adjacent to the weigher and adapted to generate an output electrical pulse when the goods are moved away from the weigher; a calculator coupled to said first and second memories for multiplying the weight of the goods with the price per unit weight; a control circuit connected to said sensing device for actuation to start a multiplication sequence when said electrical pulse is received; said calculator adapted to provide a price for the goods in digit form and to apply the price to an output circuit; and a printer coupled to the calculator output circuit for printing the price of the goods on the label.

2. A weighing and price calculating device comprising a weigher, a calculator connected to said weigher and adapted to receive signals therefrom corresponding to the weights of goods passed to said weigher, a data distributor connected to said calculator, and a printer coupled to said data distributor for printing out signalled weights and prices calculated therefrom on to labels to accompany the weighed goods; the improvement which includes a first memory connected to the weigher, for receiving a series of digits representing the weight of the goods on the weigher; a second memory in which is stored a series of digits representing the price per unit weight of the material being weighed; a third memory in which is stored a series of symbols indicating the address of a purchaser; a sensing device positioned adjacent to the weigher and adapted to generate an output electrical pulse when the goods are moved away from the weigher; a calculator coupled to said first and second memories for multiplying the weight of the goods with the price per unit weight; a control circuit connected to said sensing device for actuation to start a multiplication sequence when said electrical pulse is received; said calculator adapted to provide a price for the goods in digit form and to apply the price to an output circuit; and a printer coupled to the calculator output circuit for printing the price of the goods on a label.

3. A weighing and calculating device according to claim **1** whereby said third memory includes a series of punched data cards, said cards available for manual insertion into said printer for printing the address contained in the selected card on the goods.

4. A weighing and price calculating device according to claim **1** wherein said printer includes a slot for depositing printing plates, each printing plate carrying the name of a commodity.

5. A weighing and price calculating device according to claim **1** wherein the calculator includes two multiplying circuits which operate independently of one another, each of said circuits coupled to a comparator circuit for checking the products of the multiplying circuits.

* * * * *