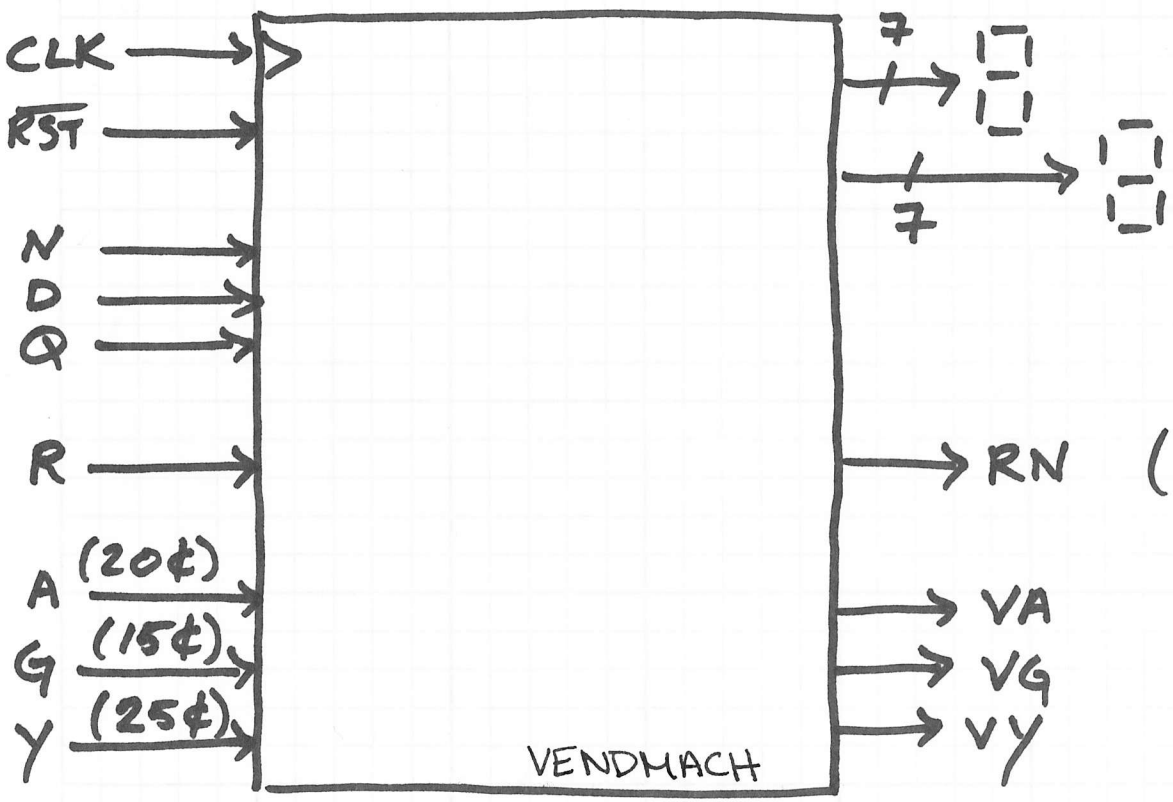
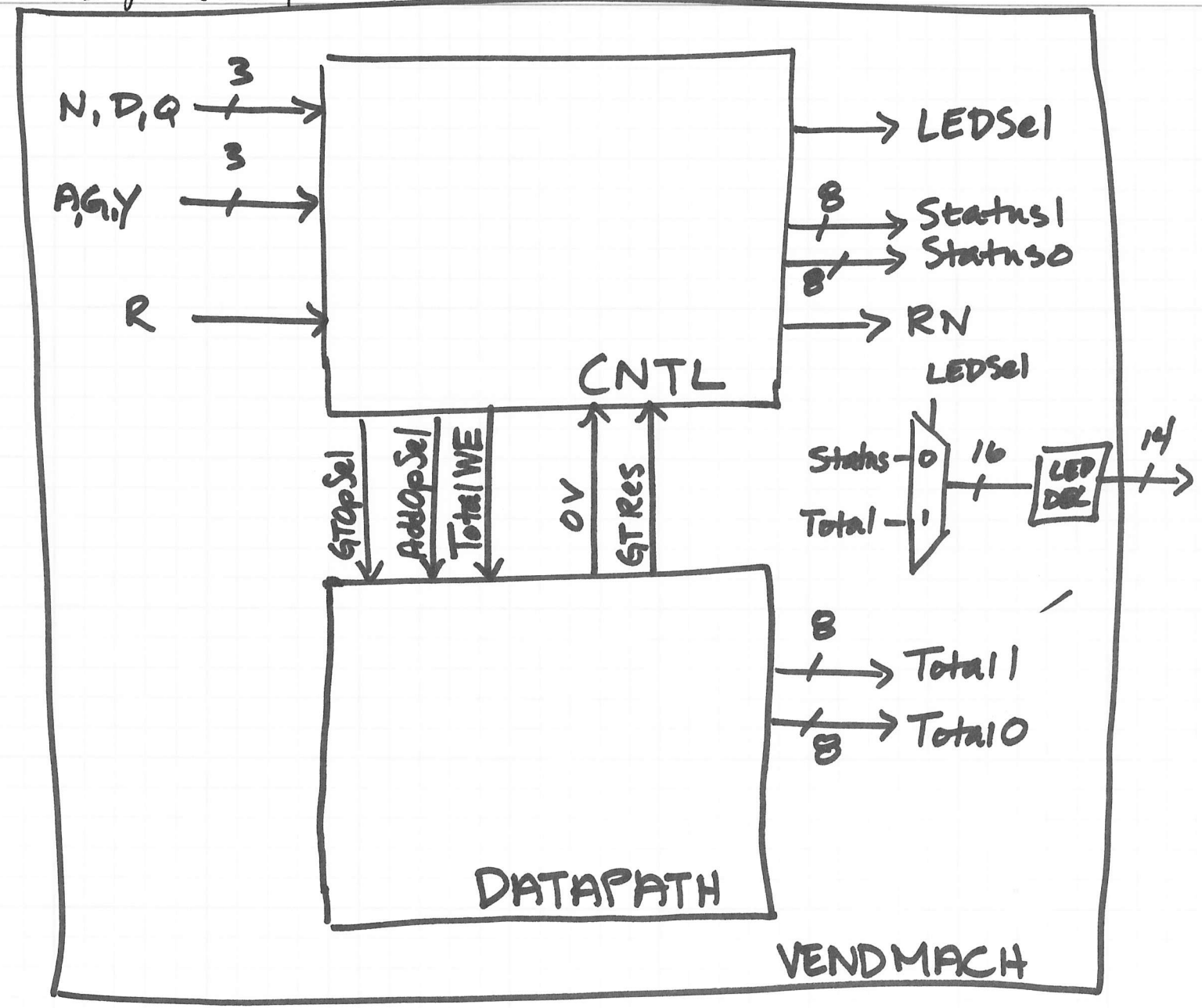
Vending machine design exercise (3/1/10)



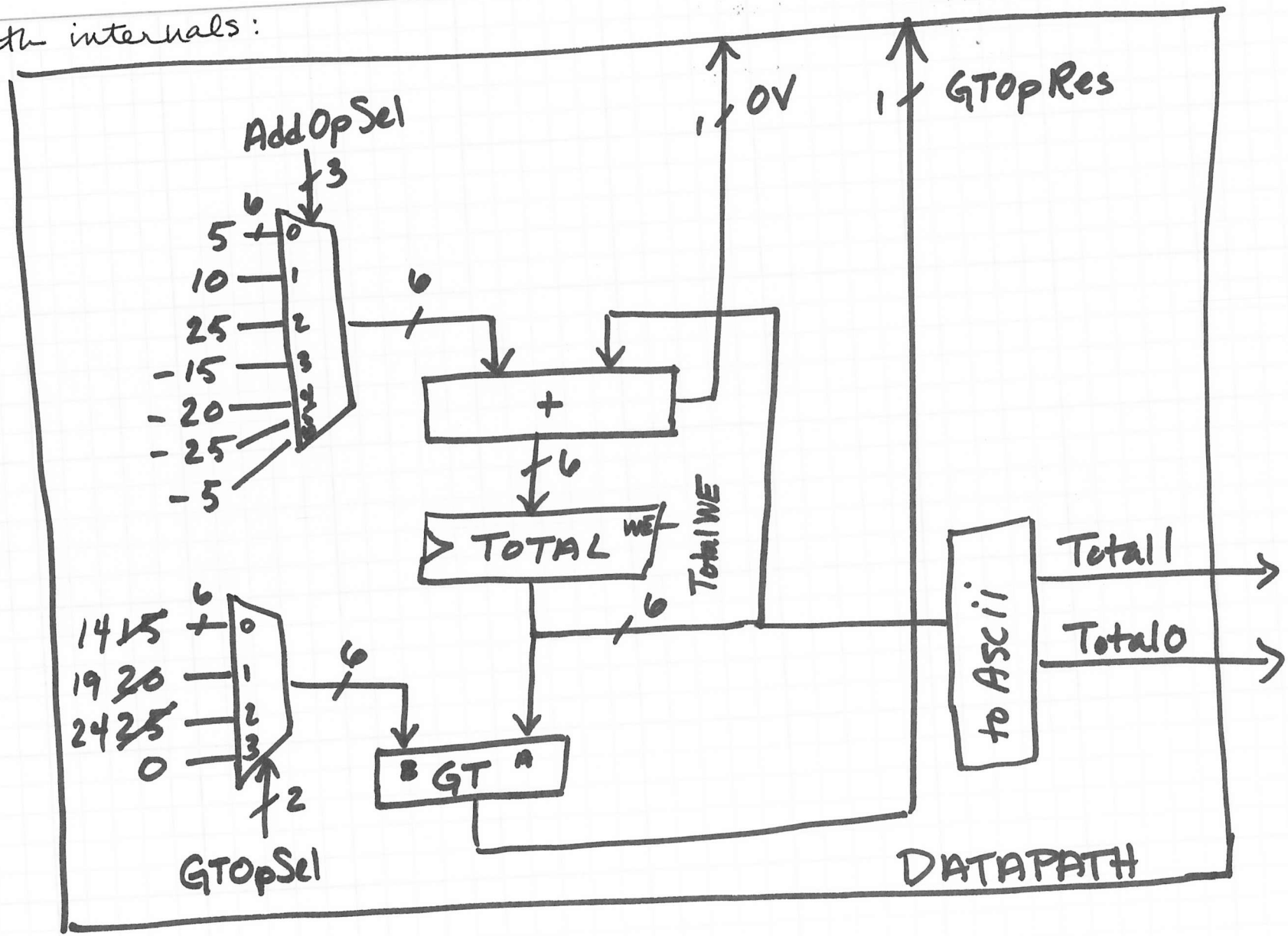
(either total sum entered, or "Er" in case of error)

(returns only nickels)

Two-part design: datapath + controller



Datapath internals:



Useful expressions for user actions (simplifies control design)

$REQA = A\bar{G}\bar{Y} \cdot NOCOIN \cdot \bar{R}$

$REQY = \bar{A}\bar{G}Y \cdot NOCOIN \cdot \bar{R}$

$REQG = \bar{A}GY$ " "

$REQ = REQA + REQG + REQY$

$NOREQ = \bar{A}\bar{G}\bar{Y}$

$NICKEL = N\bar{D}\bar{Q} \cdot NOREQ \cdot \bar{R}$

$DIME = \bar{N}D\bar{Q} \cdot$ " $\cdot$ "

$QUARTER = \bar{N}\bar{D}Q \cdot$ " $\cdot$ "

$COIN = NICKEL + DIME + QUARTER$

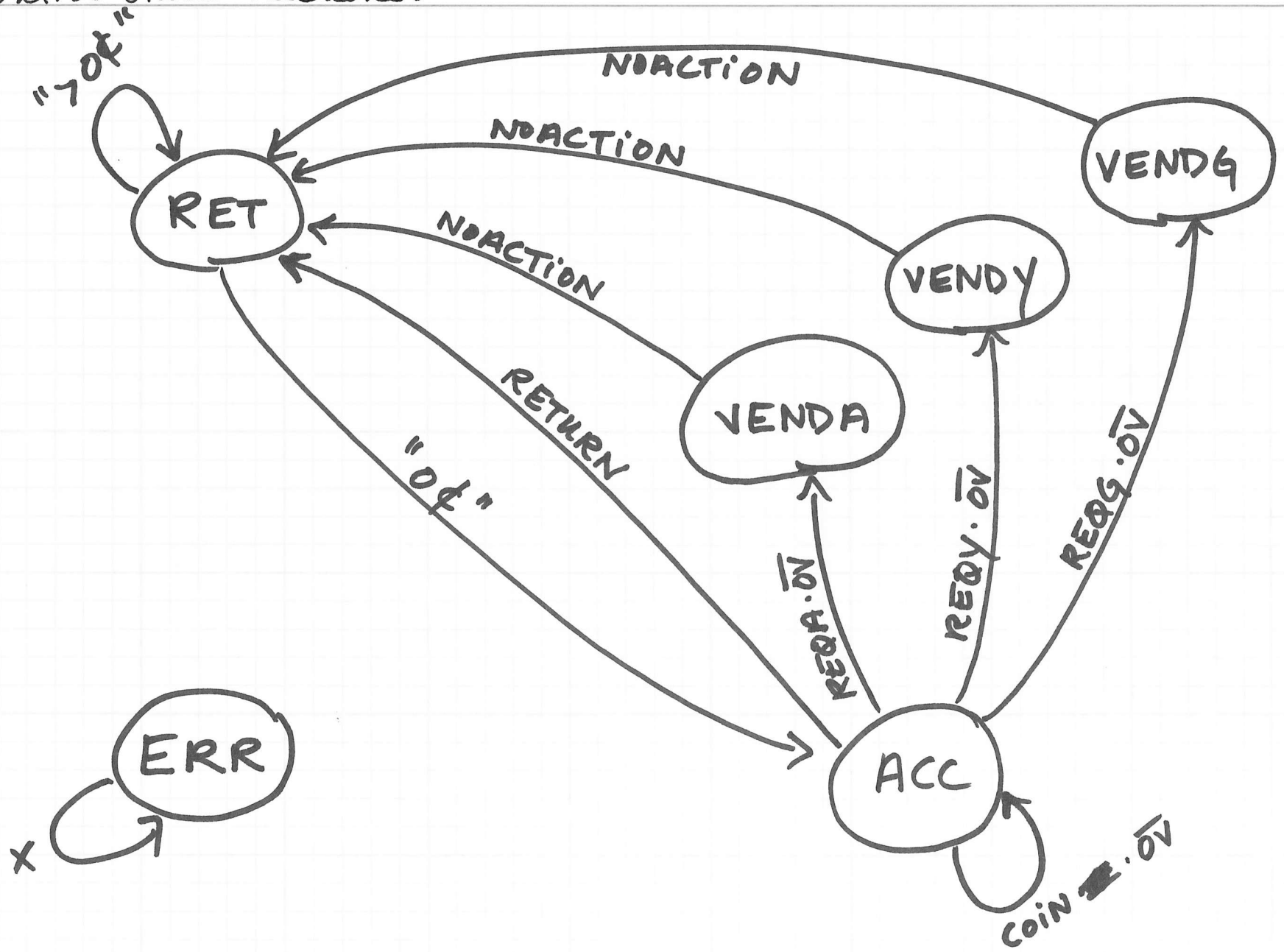
$NOCOIN = \bar{N}\bar{D}\bar{Q}$

$RETURN = NOCOIN \cdot NOREQ \cdot R$

$NOACTION = NOCOIN \cdot NOREQ \cdot \bar{R}$

legal user actions

control State machine:



Control state table:

State	UserAction	OV	GTRes	Next state	GTOpSel	AddOpSel	TotalWE	RN	VA, VY, VG	Status	LEDSel	D_2 D_1 D_0
ACC	NICKEL	0	X	ACC	XX	000	1	0	000	OK	1	000
	DIME					001						
	QUARTER					010						
ACC	REQA	0	X	VENDA	01	100	1	0	000	OK	1	001
	REQY			VENDY	10	101						010
	REQG			VENDG	00	011						011
ACC	RETURN	0	X	RET	11	XXX	0	0	000	OK	1	100
VENDA	NOACTION	0	X	RET	11	XXX	0	0	100	OK	1	100
VENDY									010			
VENDG									001			
RET	NOACTION	0	1	RET	11	000	1	1	000	OK	1	100
			0	ACC	XX	XXX	0	0				000
ERR	X	X	X	ERR	XX	XXX	X	X	000	Er	0	111

Select State encodings:

ACC = 000 (so $\overline{RST}$ will automatically start the machine in ACC state)
VENDA = 001
VENDY = 010
VENDQ = 011
RET = 100
ERR = 111

Choose flip-flop type (D), & add "next state" signals to state table ($D_2 D_1 D_0$).

Let current state be $Q_2 Q_1 Q_0$.

Define expressions for current state:

$$ACC = \overline{Q_2}\,\overline{Q_1}\,\overline{Q_0}$$

$$VENDA = \overline{Q_2}\,\overline{Q_1}\,Q_0$$

$$VENDY = \overline{Q_2}\,Q_1\,\overline{Q_0}$$

$$VENDQ = \overline{Q_2}\,Q_1\,Q_0$$

$$RET = Q_2\,\overline{Q_1}\,\overline{Q_0}$$

$$ERR = Q_2\,Q_1\,Q_0$$

Define output signals:

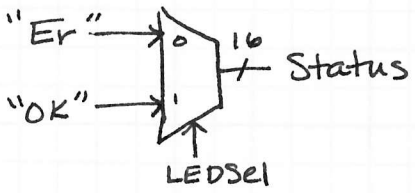
$RN = RET \cdot NOACTION \cdot \overline{OV} \cdot GTRes$

$VA = VENDA \cdot NOACTION \cdot \overline{OV}$

$VY = VENDY \cdot NOACTION \cdot \overline{OV}$

$VG = VENDG \cdot \quad " \quad \cdot "$

$LEDSel = \overline{ERR}$



$Total\,WE = ACC \cdot \overline{OV} \cdot (NICKEL + DIME + QUARTER + REQA + REQY + REQG) +$
$RET \cdot NOACTION \cdot \overline{OV} \cdot GTRes$
(the above is direct from state table, but can be simplified using summary terms defined earlier.)
$= ACC \cdot \overline{OV} \cdot (COIN + REQ) + RET \cdot NOACTION \cdot \overline{OV} \cdot GTRes$

$AddOpSel_2 = ACC \cdot \overline{OV}(REQA + REQY)$

$AddOpSel_1 = ACC \cdot \overline{OV} \cdot (QUARTER + REQG)$

$AddOpSel_0 = ACC \cdot \overline{OV} \cdot (DIME + REQY + REQG)$

$GTOpSel_1 = ACC \cdot \overline{OV}(REQY + RETURN) + (VENDA + VENDY + VENDG) \cdot NOACTION \cdot \overline{OV} +$
$RET \cdot NOACTION \cdot \overline{OV} \cdot GTRes$

$GTOpSel_0 = ACC \cdot \overline{OV}(REQA + RETURN) + (VENDA + VENDY + VENDG) \cdot NOACTION \cdot \overline{OV} +$
$RET \cdot NOACTION \cdot \overline{OV}$

Define next state signals:

$$\overline{D_2} = ACC \cdot \overline{OV} \cdot (COIN + REQ) + RET \cdot NOACTION \cdot \overline{OV} \cdot \overline{GTRes}$$

$$\overline{D_1} = ACC \cdot \overline{OV} \cdot (COIN + REQA + RETURN) +$$
$$(VENDA + VENDY + VENDG + RET) \cdot NOACTION \cdot \overline{OV}$$

$$\overline{D_0} = ACC \cdot \overline{OV} \cdot (COIN + REQY + RETURN) +$$
$$(VENDA + VENDY + VENDG + RET) \cdot NOACTION \cdot \overline{OV}$$

(Enumerate all cases in which $D_i = 0$ because we want $D_i = 1$ so that "all other" cases to go to ERR.)