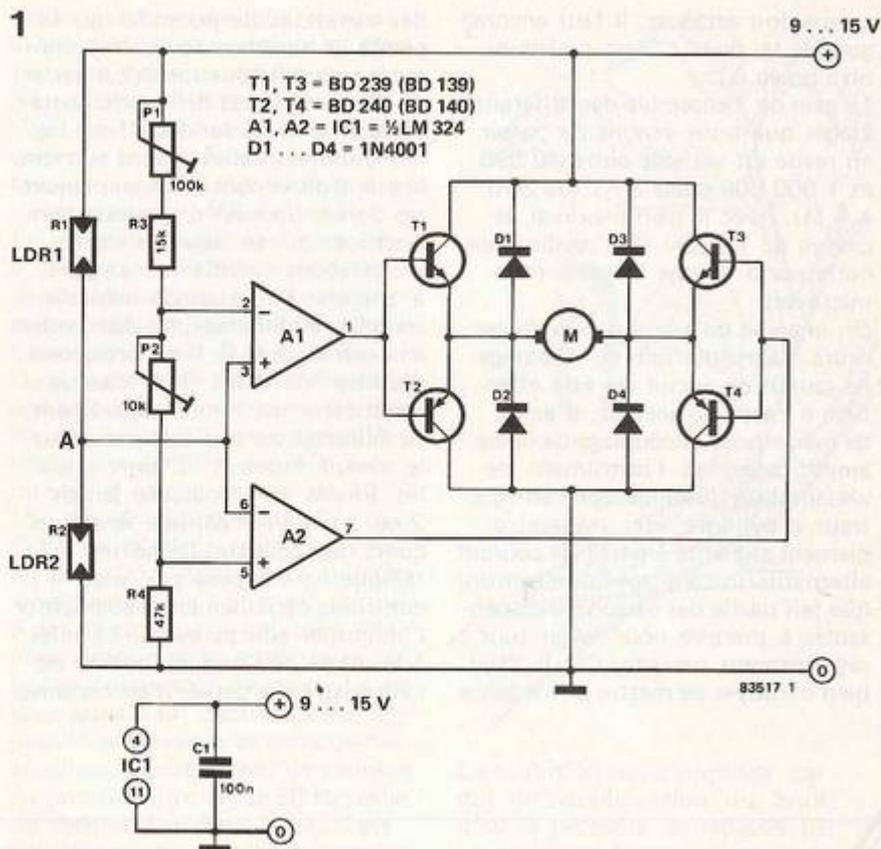


31

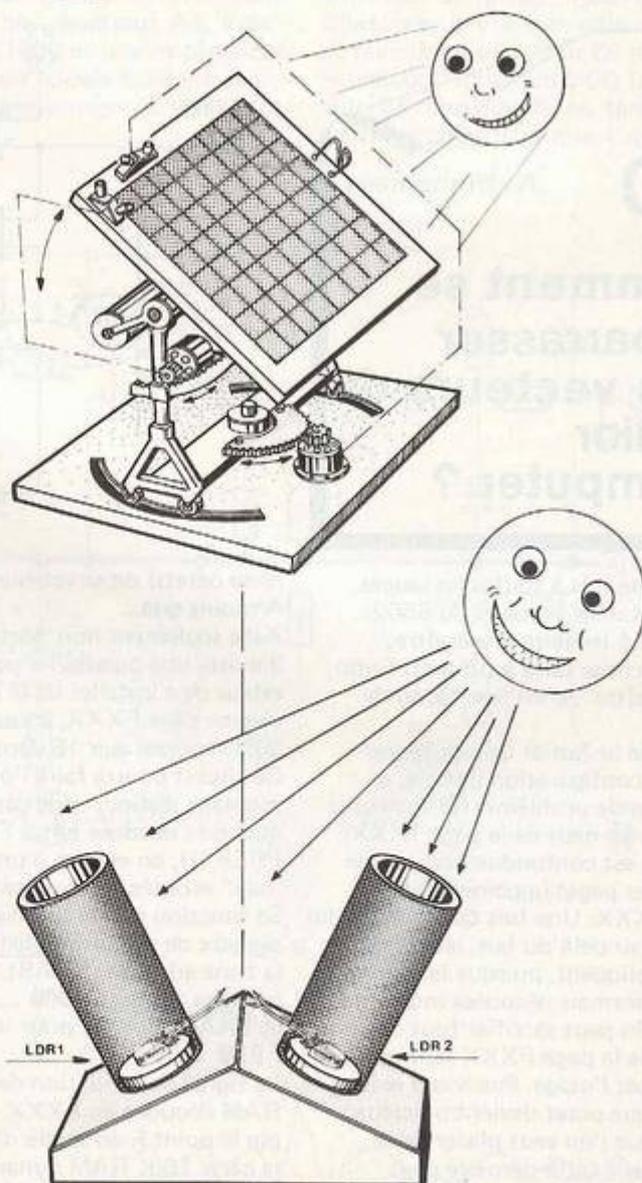
à la poursuite du soleil

On nous a beaucoup rabattu les oreilles avec des slogans du type "ne gaspillez pas l'énergie...". Permettez-nous de reprendre ces mots ici, et d'y ajouter notre grain de sel; ce qui donne: "ne gaspillez pas l'énergie *solaire* !". Mais oui, la terre tourne, le soleil se ballade d'un horizon à l'autre, et nos panneaux solaires, eux, restent la plupart du temps désespérément immobiles sur leur socle, alors que tout "photovoltaïste" averti sait que l'angle d'incidence des rayons sur les cellules est de la plus grande importance. Avec le comparateur à fenêtre proposé ici, le rendement d'une installation solaire pourra être considérablement amélioré. Si le circuit lui-même est très simple, il n'en est pas de même pour le dispositif mécanique (électro-mécanique en fait) dont la mise en place requiert quelque compétence. Le socle supportant les panneaux devra être mobile, si possible sur le plan horizontal et sur le plan vertical, ou tout au moins sur l'un des deux. Le schéma proposé n'assure la commande que d'un seul moteur, donc la mobilité sur un plan; pour le deuxième, il suffit de réaliser un autre exemplaire du même circuit. Signalons, en passant, que s'il est question de cellules solaires ici, cette application n'est pas limitative pour autant; on peut aussi se mettre "à la poursuite du soleil" pour une exposition optimale d'un cageot de tomates à mûrir...

Tant que l'éclairement des deux LDR (résistances photosensibles) est identique, le moteur est au repos: le potentiel présent sur l'entrée non-inverseuse de A1 et sur l'entrée inverseuse de A2 est égal à la moitié de la tension d'alimentation. En se déplaçant, le soleil fini par éclairer une LDR plus que l'autre (à condition qu'elles soient disposées sur deux plans sécants). La tension de l'une des entrées du comparateur à fenêtre change par rapport à celle de l'autre, de sorte que la sortie délivre une information binaire utilisée pour déterminer le sens de rotation du moteur (gauche/droite).



2



La commutation du sens de rotation est assurée par T1...T4 montés en pont. Les jonctions collecteur-émetteur sont doublées par les diodes D1...D4 dont la fonction est d'éliminer les pointes de tension pouvant apparaître lors de la mise en route du moteur.

Il y a lieu d'ajuster soigneusement P1 et P2 de telle sorte qu'à éclaircissement égal des LDR, le moteur ne tourne ni dans un sens ni dans

l'autre. Ensuite on éclaire la LDR 1 et on aveugle (partiellement) la LDR 2; le potentiel au point A doit augmenter. La sortie de l'amplificateur opérationnel A1 fournit une tension proche de la tension d'alimentation: T1 et T4 deviennent conducteurs.

Lorsque l'on inverse le rapport d'éclaircissement des deux LDR, le potentiel au point A doit chuter nettement en-dessous de la moitié

de la valeur de la tension d'alimentation, et c'est la sortie de l'amplificateur A2 qui passe au niveau haut. A présent T2 et T3 conduisent, et le sens de rotation du moteur est inversé.

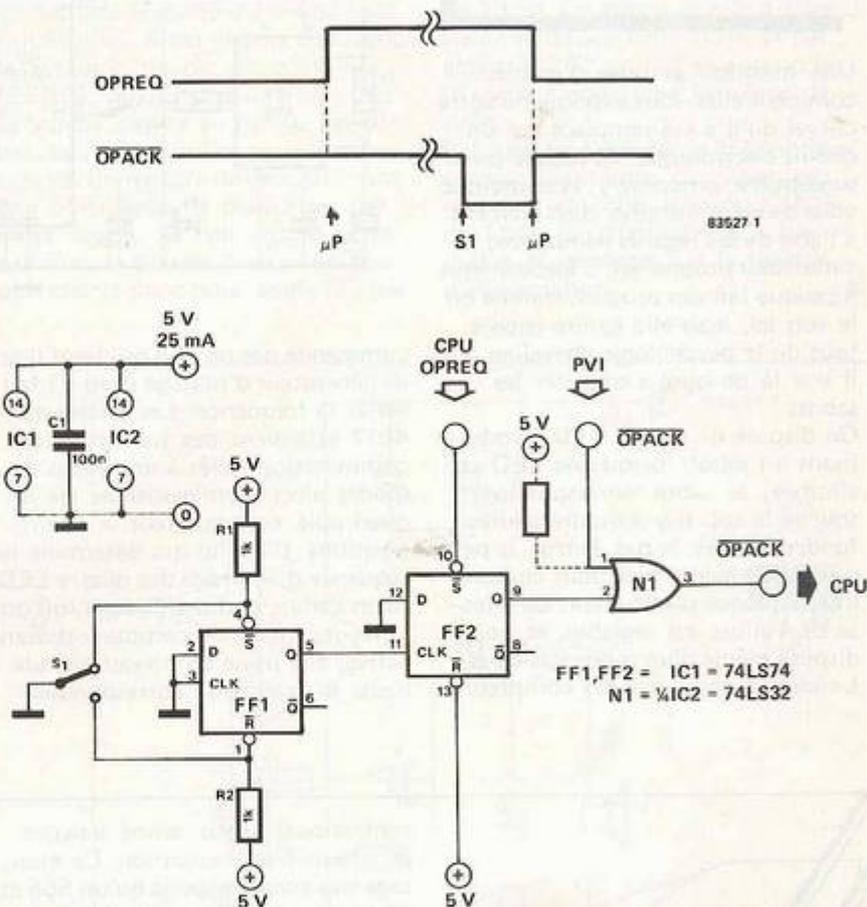
Le choix du moteur est une affaire personnelle; le courant de service maximal ne devrait pas excéder 300 mA, quelque soit le type. **M**

32

mode pas à pas pour le 2650

Pouvoir faire passer le microprocesseur en mode pas à pas lors du déverminage de programmes personnels, est l'une des aides les plus efficaces dont puisse bénéficier l'apprenti-sorcier/programmeur. Les possesseurs d'ordinateurs ayant pour cœur un 2650, catégorie dont font partie les propriétaires de l'ordinateur pour jeux TV ne pouvaient pas, jusqu'à présent, disposer de cet outil fort pratique. Et pourtant, il suffisait d'y penser, car le circuit permettant d'ajouter cette fonction supplémentaire est très simple.

Il suffit de faire croire au 2650, qu'on lui a connecté une mémoire très lente, pour le faire "marcher au pas". Dans ces conditions, les bus d'adresses et de données restent validés pendant chaque opération de lecture ou d'écriture jusqu'à ce qu'ait lieu une pression sur un commutateur. On peut ainsi vérifier pas à pas toutes les adresses et données sans risquer le blocage du système. Le circuit comprend deux bascules, un commutateur à poussoir (digitast) et une porte logique; grâce à ces divers composants et, à l'utilisation des signaux OPREQ (Operation Request) et OPACK (Operation Acknowledge), on crée un nouveau signal OPACK. Le signal OPACK d'origine est combiné à la sortie Q de FF2 par l'intermédiaire de la porte OR N1. La ligne OPACK d'origine doit être coupée, en portant son attention sur la résistance qui la force



au niveau logique haut pour veiller à ce que celle-ci reste en place du côté du PVI. A chaque cycle mémoire, la ligne OPREQ se trouve au niveau logique bas ce qui entraîne l'initialisation du 74LS74 et le passage au niveau logique haut du signal OPACK. Un nouveau cycle mémoire démarre lorsque les lignes OPREQ et OPACK se trouvent simultanément au niveau logique haut, le microprocesseur devant ensuite attendre que la ligne OPACK redescende au niveau logique bas lors d'une action sur S1. La seule fonction de FF1 est de supprimer les rebonds du commutateur à poussoir S1.

Lorsque cette modification est effectuée, on pourra s'intéresser de plus près aux bus d'adresses et de données à l'aide d'un multimètre ou mieux encore à l'aide du montage que nous venons de décrire. Lors des

interruptions, il pourrait arriver qu'il se passe des choses bien étranges avec l'ordinateur pour jeux TV. Lors de l'action sur le poussoir d'initialisation (reset), (bus d'adresses à 0000, donnée à 1F), le processeur effectue une procédure de traitement d'interruption, ce qui pourrait donner à penser qu'il y a anomalie. C'est pour cette raison, que nous disions qu'il était préférable de couper la ligne d'interruption entre l'unité de traitement centrale (CPU) et le PVI, raison pour laquelle, la résistance qui force la ligne au niveau logique haut doit rester en place du côté du CPU. Après initialisation, il sera possible de faire exécuter le programme sans le moindre problème. En ce qui concerne le bouton-poussoir S1, la meilleure solution est d'utiliser un commutateur à contact inverseur (digitast). **M**