

Az MPS-7541 kalkulátor integrált áramkör alkalmazása

DR. RYSZARD SOBKOVIK
Politechnika Poznańska
Instytut Automatyki

A korszerű elektronika a klasszikus analóg áramköri feladatokat egyre több alkalmazási területen digitális áramkörökkel oldja meg. Tipikus példa erre a mért eredmények kiértékelése, és ezzel kapcsolatban néhány egyszerű aritmetikai művelet végrehajtása (pl. szorzás, osztás, gyökvonás, négyzetre emelés, reciprokképzés, decibellszámítás stb.) Ez mindig használható abban az esetben, ha a mérési eredmények már kódolt formában állnak rendelkezésre.

Jelen cikk a zsebszámológépekhez kifejlesztett LSI integrált áramkörök alkalmazási lehetőségeit vizsgálja ezen a területen, különös figyelmet fordítva a periódusidőt mérő frekvenciamérőkben szükséges reciprokképzésre.

A feladat kitűzése

A direkt, számláló elven működő frekvenciamérés pontossága a véletlen zavaró hatásokon kívül a mérőberendezés időalapjának pontosságától és a megmért (számlált) periódusok számától függ. Ez utóbbi hibahatár csökkentése csak oly módon történhet, hogy a periódusok számát növeljük, ami sajnos a mérési idő növekedését vonja maga után. A probléma megkerülhető azáltal, hogy a frekvenciamérést periódusidő mérésre vezetjük vissza, és a periódusidőből számolással állítjuk elő a frekvencia értékét. Az eljárás előnye az, hogy a kisfrekvenciás mérések lebonyolításához szükséges idő csökken, mivel a méréshez elegendő a bejövő jel egy vagy néhány periódusának a vizsgálata.

A megoldáshoz szükséges reciprokképzést kalkulátor integrált áramkörrel valósítjuk meg. A tárgyalt elrendezés természetesen egyéb feladatok ellátására is alkalmas, hiszen a kalkulátor integrált áramkör beépített algoritmusai közül bármelyik, vagy esetleg bármelyek aktivizálása lehetséges.

Jelen cikk az MPS 7541 integrált áramkörtípus felhasználási lehetőségeit vizsgálja és olyan vezérlőrendszer ismertet, amely alkalmas a fentebb vázolt feladatok megoldására.

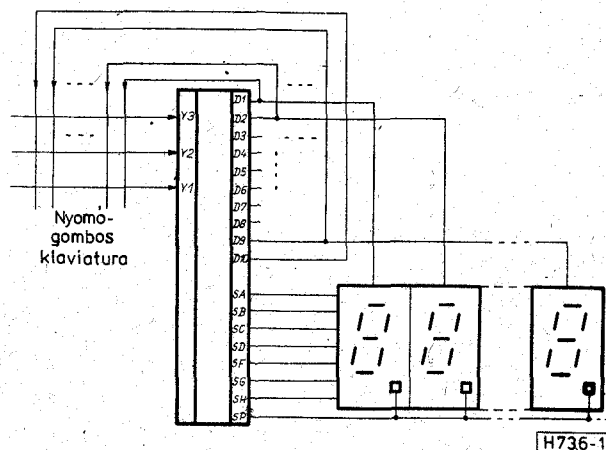
Az áramkör típusának kiválasztását a viszonylag alacsony áron és az egyszerű beszerezhetőségen kívül az is indokolja, hogy +5 V tápfeszültséggel még ki-

fogástalanul működik, így TTL elemekkel közös tápfeszültségről működtethető. Ezenkívül a kimenetén megjelenő logikai szintek TTL kompatibilisak, illetve az MPS és TTL áramkörök összekapcsolásakor fellépő illesztési problémák +5 V-ról működő CMOS inverterek segítségével egyszerűen megoldhatók, mivel egy inverter kimenetére két TTL kapu bemenete kapcsolható.

Az MPS 7541 áramkör vezérlőrendszere

A téma részletesebb tárgyalása előtt ismertetjük az MPS 7541 típusú kalkulátor vezérlési lehetőségét és a vezérléshez szükséges áramkörrel szemben támasztott igényeket. Az MPS áramkör pozitív logikai rendszerben dolgozik.

A kalkulátor integrált áramkör vezérlési rendszerének a vázolata az 1. ábrán látható. A kijelzési ciklusok során a D1, D2, ..., D10 kimeneteken egymást követően logikai 1-es jelenik meg. Ez lehetővé teszi a számjegyek ciklikus kijelzését oly módon, hogy a D1, D2, ..., D10 kimenetek sorra engedélyezik a számkijelzőket, az SA, SB, ..., SP kimeneteken megjelenő kombinációk pedig döntenek az egyes számjegyekről. A tizedespont megjelenítéséről az SP kimenet gondoskodik. Egy kijelzési ciklus időtartama kb. 12 ms, így minden 12 ms-os ciklusban egy-egy számjegy 1,2 ms időtartamon keresztül világít.



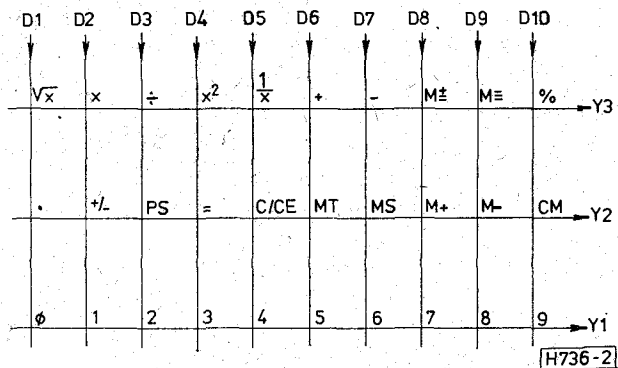
1. ábra. Az MPS 1541 vezérlési rendszerének a vázolata

Beérkezett: 1980. április 16.

Az adatbetáplálás az Y1 bemeneten történik, a műveleti kódokat pedig az Y2 és Y3 bemenetpár fogadja. Egy számjegyet akkor fogad el az egység, ha a D1, D2, ..., D10 kimenetek és az Y1 bemenet közötti összeköttetés legalább négy kijelzési ciklusideig fennáll. A D1, Y1 összekapcsolásával a „0”, a D2, Y1 kombinációval az „1”, a D10, Y1 összekapcsolásával pedig a „9” számjegy vihető be a kalkulátor memóriájába.

A műveletek végrehajtásának kezdeményezésekor a D1, D2, ..., D10 kimeneteket az Y2 vagy Y3 bemenetekkel kell összekapcsolni. A 2. ábrán összefoglaltuk az egyes kombinációkhoz tartozó műveleti kódokat.

Megjegyezzük, hogy két adat vagy utasítás bevitel között is legalább négy ciklusidő hosszúságú szünetet kell tartani. Ilyenkor a Y bemenetek egyike sem lehet aktív.



2. ábra. Az MPS 1541 bemeneti jelkombinációi

A kifejlesztett vezérlő áramkör ismertetése

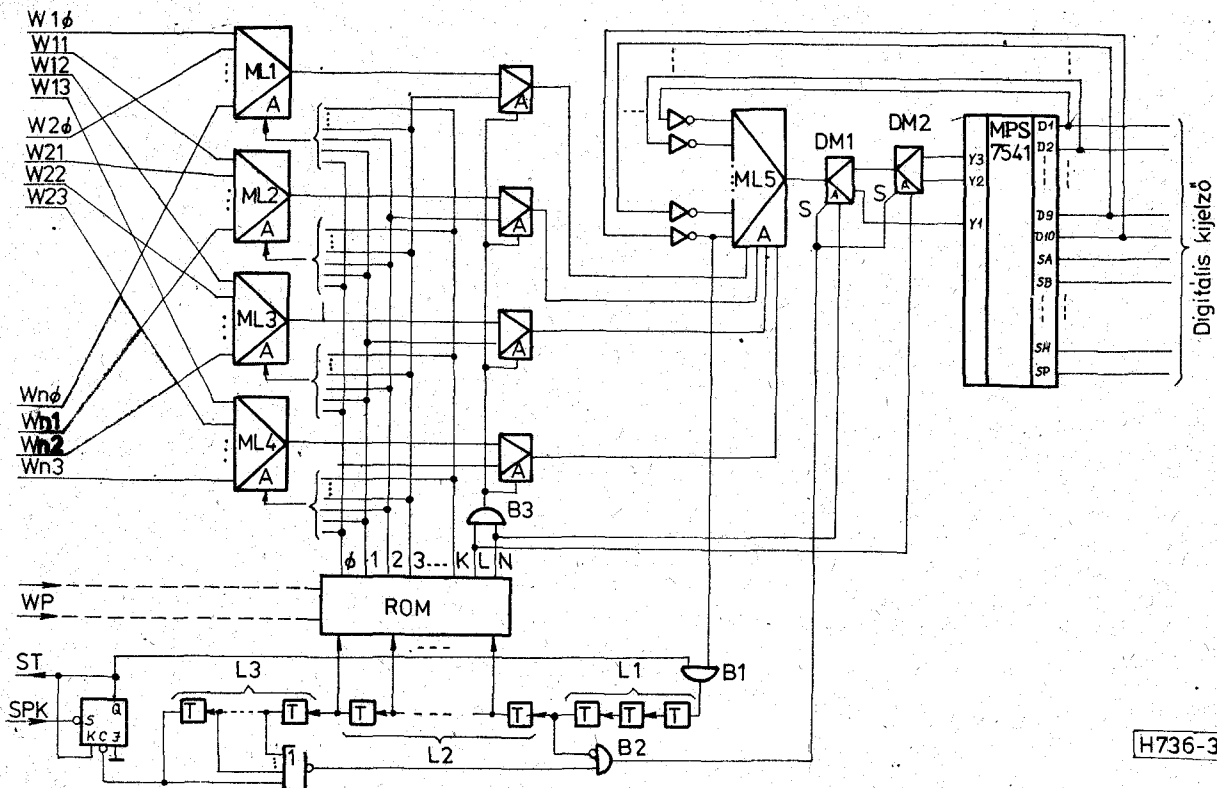
Az általunk kifejlesztett berendezésben a kalkulátor integrált áramkört nem kézi billentyűzettel, hanem egy speciális szinkron vezérlőegységgel működtetjük, amely logikai „1”-est ad az Y1 bemenetre, ha számot táplálunk az eszköz memóriájába, és ugyanígy vezérli az Y2, Y3 bemenetet, ha valamilyen műveleti kód beviteléről van szó. Természetesen elektronikus vezérlés esetén is be kell tartani az egyes utasítások közötti szünetre vonatkozó előírásokat. Ilyenkor az Y1, Y2, és Y3 bemenetek közül egyiken sem lehet logikai „1”-es.

A vezérlőegység kapcsolási rajza a 3. ábrán látható. Ez az áramkör lehetővé teszi azt, hogy a kézi klaviatúra összes lehetséges utasítását elvégezhessük.

Az áramkör akkor kezd működni, ha a tároló Q kimenetén logikai „1” jelenik meg az SPK jel hatására. Az SPK a mérőegység felől érkezik, és egyértelműen jelzi, hogy a mérés (esetünkben a periódusidő mérése) befejeződött. Az MPS áramkör munkaciklusának befejezése után az L1 számláló egyet lép, mivel a B1 kapu nyitott állapotban van. A mérőberendezés felől BCD kódban érkező adatok az ML1, ML2, ML3 és ML4 multiplexerek W1Ø – Wn3 bemeneteire érkeznek. A bemeneteket n darab négyes csoportba osztottuk, és a csoportok egy-egy dekádnak felelnek meg. Mindegyik multiplexer n bemenettel rendelkezik, és a megfelelőt ezek közül a ROM memória Ø-k kimenete választja ki. Ily módon a ROM aktív kimeneteinek száma és a dekádok száma között a

$$2^k \geq n$$

kapcsolat áll fenn.



3. ábra. A megvalósított áramkör kapcsolási rajza

Az áramkör működésének lépései során tehát az adatok és a műveleti kódok a kalkulátor áramkörbe jutnak. Egy lépés időtartama az MPS áramkör 8 munkaciklusának az időtartamával azonos. Ezt az L1 számláló biztosítja. (L3 nulla állapotban van.) Egy lépés tehát 8 ciklusra bontható. Az első négy ciklus lefutási ideje alatt a B2 kimenetén levő S jel logikai „1”-gyel vezérli a DM1, DM2 demultiplexereket. Ekkor az Y1, Y2, Y3 bemeneteken keresztül az adatok és utasítások bejutnak a kalkulátor memóriájába. A következő négy ciklusban S logikai „0”, így az Y1, Y2 és Y3 bemenetekre nem jut információ (ez biztosítja a vezérlések közötti szüneteket). Az egyes lépések során a ROM kimenetén megjelenő kombináció dönt a realizálandó műveletről, a következő lépés kiválasztása pedig az L2 számláló segítségével történik. Az L2 számláló hossza meghatározza a művelet elvégzéséhez szükséges lépések számát. A memória N rekeszében pedig az az információ található, hogy az adott lépés során az MPS-be számot írunk-e be, vagy műveletet választunk-e ki, ugyanis az N kimenet a DM1 demultiplexert vezérli. A DM1 bemenete az ML5 multiplexer kimenetére kapcsolódik, így ha $N=1$ vagy $N=0$ akkor az Y1, Y2, Y3 bemeneteken keresztül adatok vagy műveleti kódok kerülhetnek a kalkulátor egységbe. Az ML5 címbemenetein (A) levő jelek kombinációja meghatározza azt, hogy az MPS munkaciklusának ideje alatt mikor jelenik meg a logikai „1”-es a multiplexer kimenetén. Ha $N=1$ és a $S=1$ akkor az MPS-be olyan szám kerül, mely azonos az ML5 címbemenetein levő BCD kódban írott számmal. Ilyenkor ugyanis az adatbemeneteken kiválasztott és a D1, D2, ..., D10 kimenetekről érkező információ az Y1 bemenetre kerül (1. ábra).

Ebben az esetben a ROM L jelzésű kimenete azt dönti el, hogy adott lépésnél az ML5 címbemeneteit a ROM $\emptyset - 3$ kimeneteiről, vagy az ML1–ML4 multiplexerek kimeneteiről vezéreljük.

$L=0$ és $N=1$ esetében a $B3=0$, így az MPS memó-

riájába kerül a ROM $\emptyset - 3$ kimenetein levő kódolt számjegy. Ezt akkor használjuk, ha konstansokat olvasunk ki a memóriából. A másik esetben ($L=1$), amikor a B3 kimeneten logikai „1” jelenik meg, a kalkulátor memóriájába az ML1–ML4 kimenetein levő kódolt számjegy kerül.

Ha a ROM N kimenetén logikai „0” jelenik meg, akkor az ML5 kimeneti jele a DM2 bemenetére jut, és ezen a demultiplexeren keresztül az Y2 vagy Y3 bemeneteket vezérli. A két lehetséges bemenet közül a szükségeset a ROM L kimenetének logikai állapota választja ki. Ugyanakkor a ROM $\emptyset - 3$ kimenete meghatározza azt is, hogy a ciklus melyik fázisában jelenik meg logikai „1”-es az ML5 kimenetén. Így a memória $\emptyset - N$ állapota egyértelműen meghatározza a kalkulátor által elvégzendő feladatokat.

Ha a működés során az L2 számláló ismét nulla állapotot vesz fel, az azt jelzi, hogy az összes adat- és műveleti kód a memóriába került. Ilyenkor az L3 számláló „1”-be billen és a B2 jel a DM1, DM2 demultiplexereket nullázza. Ily módon a kalkulátor Y1, Y2 és Y3 bemenetei a vezérlőegység állapotától függetlenül logikai „0” jelet kapnak.

Ez alatt az idő alatt az áramkör az éppen befejezett lépés eredményeit írja ki és nem realizálja a soron következő lépéseket mindaddig, amíg L3 számláló a tárolót alapállapotba nem billenti. Ezután a rendszer újra indítható.

Ha több programot akarunk megvalósítani, akkor ehhez a memóriában nagyobb területet kell kijelölnünk. A megfelelő program kiválasztása a ROM WP címekre adott számkombinációval történhet.

*

A szerző köszönetet mond a BME HEI kollektívájának, köztük elsősorban Vannai Nándor és Pap László adjunktusoknak az Intézetben végzett kutatás során nyújtott támogatásért.