

UŽIVATELSKÁ PŘÍRUČKA

OSOBNÍ MIKRO POČÍTAČ PMD-85

III. OPERAČNÍ SYSTÉM



TESLA - ELEKTRONICKÉ SOUČÁSTKY

OSOBNÍ MIKROPOČÍTAČ PMD-85

UŽIVATELSKÁ PŘÍRUČKA

III. OPERAČNÍ SYSTÉM



TESLA - ELEKTRONICKÉ SOUČÁSTKY

POPIS ZAPOJENÍ - OPERAČNÍ SYSTÉM

OBSAH

Úvod	1
I. Popis koncepce	1
II. Hardware systému	4
1. Modul mikropočítače s videoprocesorem	4
1 a. Centrální jednotka mikropočítače	4
1 b. Paměťový podsystém	4
1 c. Videoprocesor	6
1 d. Logika interfejsu klávesnice	8
1 e. Vstupně-výstupní strana mikropočítačové desky	9
2. Klávesnice	9
3. ROM modul	11
4. Modul interfejsů	12
Paralelní programovatelný VSTUP/VÝSTUP	12
Interfejs IMS - 2	13
Sériový styk V24	13
Interfejs pro kazetový magnetofon	13
Časovač	14
Vstupně-výstupní aplikační konektor	14
III. Operační systém	15
Skladba operačního systému	17
1. Monitor	17
Podprogramy modulu monitor	21
2. EDIT	24
3. KLÁV	26
4. PRTOUT	27
5. INPOL	28
6. OSIO	29
Programové klíče	30
Modifikace systému	30

Obrázkové přílohy

Mapa paměti	obr. č. 1
Základní časová spolupráce videosystému a mikro- počítače	obr. č. 2
Formát videoúdaje	obr. č. 3
Koncepce systému PMD - 85	obr. č. 4
Schéma zapojení centrální jednotky mikropočítače.	obr. č. 5
Tabulka určující tvorbu signálu READY	obr. č. 6
Schéma zapojení paměťového podsystému RWM	obr. č. 7
Schéma zapojení paměťového podsystému ROM	obr. č. 8
Schéma zapojení videoprocessoru	obr. č. 9
Spolupráce videosystému s uP během zápisu do paměti	obr. č. 10
Spolupráce videosystému s uP při čtení z pamě- ti RWM	obr. č. 11
Schéma zapojení sérioparalelního registru video- údaje	obr. č. 12
Schéma zapojení interfejsu klávesnice	obr. č. 13
Vstupně-výstupní konektory mikropočítačové desky.	obr. č. 14
Schéma zapojení klávesnice	obr. č. 15
Pohled na klávesnici	obr. č. 16
Výběr interfejsových obvodů	obr. č. 18
Schéma zapojení paralelního interfejsu (GPIO)....	obr. č. 19
Schéma zapojení interfejsu IMS - 2 (HP-IB)	obr. č. 20
Schéma zapojení interfejsu V24	obr. č. 21
Schéma zapojení interfejsu pro magnetofon	obr. č. 22
Typické průběhy interface pro MG	obr. č. 23

Schéma zapojení interfejsu časovače	obr. č. 24
Aplikační konektor	obr. č. 25
Rozložení paměti RWM	obr. č. 27
Formáty organizování video	obr. č. 28
Vstup-výstup na OS	obr. č. 30
Ukazatele hodnoty pro modifikaci systému	obr. č. 31

UVOD

Mikropočítač PMD 85 se svou architekturou a výkonností řadí mezi osobní mikropočítače (personal computers). Je orientován na mikroprocesorový systém 8080.

Jeho hlavní předností je grafické zpracovávání údajů pod vyšším programovacím jazykem BASIC GRAPHICS (dále jen BASIC - G). Obvodová architektura je minimalizována a představuje především podporu pro programové vydavení (software). Tato strategie umožňuje uživateli přemodifikovávat systém s požadovanými vlastnostmi.

V následujících kapitolách bude uživateli podrobně vysvětlena činnost mikropočítače, ke které je nutná znalost jednotlivých stavebních prvků mikroprocesorové rodiny 8080. Tato příručka patří mezi základní uživatelské příručky, na které navazují programovací příručky jednotlivých programovacích jazyků (BASIC - G, BASEX, ASSEMBLER) a bulletiny k interfejsovým modulům.

I. POPIS KONCEPCE

Základní myšlenka spočívá v simultánní činnosti videosystému a mikropočítače s paměťovým podsystémem.

Paměťový podsystém je po 16 k byte stránkách obr. č. 1. Jedna stránka je pevně definována jako stránka paměti ROM pro firmware. Poslední stránka paměťové rodiny je vymezená jako "VIDEO PAGE", která obsahuje zobrazované údaje. Každému zobrazenému bodu přísluší jeden bit ve videostránce. Takovému rozsáhlému paměťovému podsystému konstrukčně nejlépe vyhovují dynamické paměti MHB 4116. K zobrazovaným údajům (byte) musí mít přístup videosystém i mikropočítač a z toho důvodu je paměť

RWM realizována s dvoubránovým adresovým přístupem. Nutné obnovení údajů je vyřešeno nepřetržitým čtením údajů z videostránky a jejich zobrazováním na TV obrazovku. Tuto činnost zabezpečuje videosystém. Základní časová spolupráce mezi videosystémem a mikropočítačem je znázorněna na obr. č. 22. Z něho je zřejmé, že pro simultánní činnost má každý systém vymezenou svou dobu, během které je "majitelem" paměťového podsystému RWM, případně lokální sběrnice počítače. V případech, že by došlo ke konfliktním situacím, např. když požadují oba systémy spolupráci nad pamětí RWM, ponechává se priorita systému VIDEO a mikropočítač během této doby musí být v režimu WAIT. K těmto situacím dochází zřídka a programová propustnost mikropočítače je snížena asi o 8 %.

Způsob zobrazování bodů na TVP je následující:

- jak z obr. č. 2 vyplývá, dochází k překrývání činnosti VIDEO a mikropočítače, kdy během doby určené pro VIDEO se připraví data pro přečtení ve videostránce a během doby určené pro mikropočítač dochází k jejich zobrazování. Přečtený údaj z paměti videostránky má formát znázorněný na obrázku č. 3. Z tohoto je vidět, že během dvou dob dojde k zobrazení šesti bodů, které mohou mít barevný kód uložen ve dvou bitech. Tento formát byl zvolen i z důvodu lepší manipulace při alfanumerickém režimu, kde pět bitů je informačních a jeden bit je pro meziznakovou mezeru. Při použití základního kmitočtu 18,432 MHz vychází doba zobrazování jednoho údaje (byte) 1 μ s. Pro zobrazení 48 znaků ASCII na jeden řádek je potřeba 288 bodů během 48 μ s aktivního řádkového světelného paprsku, takže zbylých 16 μ s je čas řádkového zatemňovacího impulsu.

Pro jednoduchost logiky rozkladu obrazu je použito nepřekládané řádkování a tedy z počtu 312 řádků je aktivních 256 řádků. Celkově je možno shrnout, že aktivní zobrazovací plocha

je 288 x 256 bodů, což znamená přibližně 12 k byte s videostránky.

Z předcházejícího je tedy zřejmé, že videosystém neobsahuje žádnou logiku pro dekódování příslušných znaků, ale pouze jednoduchý paralelně-sériový registr, který převeze údaje z lokální sběrnice systému a v patřičných časových impulsích jej sériově vysílá jako modulační videosignál. Dochází tedy k tzv. kopírování obsahu videostránky. Úloha mikropočítače je tedy pracovat nad tímto obsahem, modifikovat ho podle daných procedur. Jde tedy výlučně o programovou záležitost. Systém potřebuje minimální software pro svou práci, obsahující procedury pro zabezpečení nejnižších úkonů. Takovému programovému modulu říkáme firmware a v systému PMD 85 je v rozsahu 4 k byte případně je ho možno rozšířit na 8 k byte (s podporou ROM MHB 8716).

Jakoukoli změnu v organizaci systému je nutno implementovat v tomto firmware, případně, jak bude dále popsáno, přemodifikováním patřičných ukazatelů v jeho zápisníku.

Popsaná koncepce poskytuje možnost použití systému pro klasický - alfanumerický systém, kterému je plně k dispozici 48 k byte operační paměti.

Z konstrukčního hlediska byla zvolena koncepce znázorněná na obr. 4, z kterého vyplývá, že systém obsahuje centrální počítačový blok, zabezpečující všechny požadované úkony pro interaktivní zobrazovací jednotku. Tento blok je umístěn na jedné desce. Na ni navazuje výstup na TVP, pro klávesnici a vstup externích napájecích zdrojů. Mikropočítačový systém má vyvedenu sběrnici pro vstupně/výstupní podsystém. Přes tento kanál je možné rozšiřovat systém do libovolné šířky - ROM moduly, interfejsy apod.

Nejrozšířenější interfejsy mikropočítače jsou umístěny na jedné desce, která se nachází v samotném mikropočítači,

přičemž do druhé pozice je možné vkládat paměťové moduly.

II. HARDWARE SYSTÉMU

Mikropočítač PMD 85 obsahuje tyto části:

- modul mikropočítače s videoprocesorem
- modul klávesnice
- modul ROM
- modul interfejsů

Cílem této kapitoly je vysvětlit činnost jednotlivých částí.

1.a. Centrální jednotka mikropočítače

- obr. č.5 je tvořen klasickou architekturou, obsahující CPU MHB 8080A, podpůrné obvody MH 8228 a MH 8224.

Přerušovací systém je jednoúrovňový přes vektor RST 7, což umožňuje použití i kaskádního řazení obvodů MH 3214, které se připojí přes externí vstupně-výstupní konektor systému. Manuální nulování (RESET) systému je pomocí tlačítka umístěného na klávesnici a připojeného na vstup RESIN obvodu MH 8224.

Zvláštností tohoto bloku je tvoření signálu READY, který má za úkol synchronizovat mikropočítač s videoprocesorem. Tento signál je generován logikou, jejíž pravdivostní tabulka je na obr. č. 6. Jak bude dále vysvětleno, je tento signál závislý na systémovém signálu VIDEO, který určuje právě platného majitele sběrnice a na stavu, zda se mikropočítač připravuje k zápisu údaje nebo čtení údaje. Tento stav získá ze stavového slova, a to konkrétní linky D1, která udává, že bude probíhat zápis do paměti (WO).

1 b. Paměťový podsvytém

- skládá se ze dvou typů pamětí RWM a ROM, které mají vlastní

dekodér paměťových stránek. Paměť RWM - obr. 8. 7 je realizována na 24 obvody MHB 4116 a má dvoubránový adresový přístup realizovaný obvody 74153, které jsou ovládány signály z videoprocessoru, signál VIDEO přepíná "bránu" adresy buď ze systémové sběrnice mikropočítače nebo z videoprocessoru, konkrétně z čítače rozkladu obrazu. Signál AMUX má funkci přepínat dolní a horní část adresy paměti (řádek - sloupec).

Dále je nutno upozornit na signál MW, který je inicializován na úroveň +5V přes odpor 800 Ω . Tento odpor má značný význam, protože musí udržovat tuto linku na úrovni LOG 1 při vzájemném přepínání spolupráce mikropočítač a videoprocessor.

Paměť ROM - obr. 8 se skládá ze čtyř obvodů MHB 8708, ve kterých se nachází firmware systému, jsou adresovány od stránky 8000 H. Paměti RWM a ROM mají samostatné adresové (výběrové) dekodéry, obvody MH 3205 a navíc logiku pro vnucení stránky "VIDEO PAGE", která na základě signálu VIDEO aktivuje signál CAS 3 při současném aktivování signálu CAS. Pro druhý případ, jedná-li se o dobu mikropočítače, uplatňuje se výběr paměťové stránky na základě adresových linek A14 a A15 mikropočítače, za předpokladu, že klopný obvod R - S to umožňuje.

Funkce klopného obvodu umožňuje zablokovat při tzv. studeném startu (RESET) nultou stránku, která je vnucena mikroprocesorem na adresové linky sběrnice a zablokovat adresový výběrový dekodér pro paměť ROM (stránka 8000 H). Zrušení tohoto stavu klopného obvodu se provede signálem I/OW, který je generován instrukcí OUT, potom, když dojde k programovému ošetření registru PC (programový čítač) v mikroprocesoru instrukcí JMP.

Pro správnou spolupráci mikropočítače a videoprocessoru je na lince vstupu E3 dekodéru paměti RWM připojen kondenzátor cca 220 pF, který v konečném důsledku zpožďuje signál CAS 0 + 3.

1 c. Videoprocessor obr. č. 9

- jeho úlohou je vytvářet směs časových signálů pro paměťový podsystém RWM a video. Mechanismus videosystému musí zabezpečit přečtení údaje z videostránky, uložení údaje do posuvného paralelněsériového registru pro modulaci videosignálu. Z hlediska systémové sběrnice videoprocessor představuje aktivní člen pouze jako "posluchač" (listener). Je realizován logikou TTL a skládá se z těchto částí:

- část generující adresu zobrazovaného bodu

Je realizována čtyřmi čítači MH 7493. Z nejnižšího stupně čítače, který je rozdělen na modulo 2 a 5, je čítač modulo 2 použit na tvorbu signálu VIDEO, který určuje právě platného "majitele" systémové sběrnice. Čítač rozkladu obrazu je inkrementován signálem STB, který se vyskytuje pouze jednou za 1 μ s.

Výstupy z čítače jsou vedeny do adresového multiplexoru paměti RWM a do jednoduché kombinační logiky, která vytváří pro TVP synchronizační a zatemňovací impulzy (SO, SR, ZAT).

Posun řádkového synchronizačního impulzu a tedy posuv obrazu na stínítku obrazovky je možné převést přepojením přepojek.

- logika řízení dynamických pamětí

- a spolupráce s mikropočítačem, která generuje posloupnost signálů RAS, AMUX, CAS a signály pro zobrazení údaje STB a impulzy pro posun video údaje - CLOCK.

Všechny signály jsou tvořeny ze základní frekvence mikropočítače 18, 432 MHz. Touto frekvencí jsou posunovány impulzy Φ 2 (TTL) v osmibitovém sérioparalelním registru. Na každém výstupu registru vznikají dva posunuté impulzy Φ 2 (TTL), ze kterých se kombinační logikou odvádějí řídicí signály (RAS, CAS, CLOCK,

STB, AMUX). Časový sled signálů je znázorněn na obr. č. 2.

Činnost videoprocessoru ve spolupráci s mikropočítačem vystihuje obr. č. 10 a 11. Jednotlivé signály mají tento význam :

- $\Phi 1$ a $\Phi 2$ jsou hodinové impulzy μP
- \overline{MR} a \overline{EW} čtecí a zápisový signál paměti μP
- READY - signál připravenosti podsystému pro μP
- \overline{RAS} , \overline{AMUX} , \overline{CAS} - řídicí signály pro paměť MHB 4116
- VIDEO signál vymezující právě probíhající cyklus (dobu) pro videoprocessor - VIDEO reprezentuje dobu pro mikropočítač.
- STB - zápisový signál pro video - údaje do posuvného výstupního registru
- CLOCK - jsou posuvné impulzy video - údaje ve výstupním registru (2 x MH 7496)
- STSTB - signál μP pro uložení stavového slova

Jak je zřejmé z průběhů, činnost videoprocessoru je stereotypní a není měněna žádnou událostí. V cyklu videoprocessoru dochází k zablokování (třetí stav) datové a řídicí sběrnice mikropočítače a v daném okamžiku se na datové sběrnici objeví video - data, adresované v čítacím rozsahu obrazu. Signálem STB se tato data přepíše do sériového výstupního registru (obr. č. 12) a dva bity do klopného obvodu MH 7474, který slouží jako atributy barevného kódu (provedení výstupu RGB) nebo jako kódy pro snížení jasu na 50 % nebo jejich vzájemná kombinace u výstupu pro běžný černo - bílý TVP.

Je třeba ještě zdůraznit, že výstupní posuvný registr, který posouvá modulační videosignál, je náběžnou hranou signálu STB resetován, protože obvody MH 7496 umožňují pouze zápis z úrovně $\log 0$ do $\log 1$. Toto umožňuje hradlo NAND, do kterého vstupuje uvedený signál STB a výstupní signál registru MH 7496, který

nalude právě v tomto okamžiku úroveň log 1. Nulový signál je velice krátký, protože daný výstup z registru je inicializován úrovní log 0 pin č. 2/. Vraťme se k popisu spolupráce videoprocessor - mikropočítač. K zápisu údaje mikroprocesorem do paměti, který se uskutečňuje v jeho pracovním taktu T3 strojového cyklu je rozhodující, jaký cyklus byl ve stavu T2, kdy může mikroprocesor obdržet avíz o (signál READY) pro vložení prázdného taktu (TW). V případě, že v kritickém čase (viz obr. č. 10) je v tomto taktu cyklus VIDEO, následující takt bude T3, ve kterém může dojít k zápisu dat do paměti. V opačném případě bude uložen mezi T2 a T3 jeden TW a tím bude zabezpečena správná synchronizace dvou systémů nad paměťovým podsystémem. Mikroprocesor testuje signál READY sestupnou hranou impulzů $\Phi 2$. Pokud má signál READY úroveň log 1, pokračuje dalším pracovním taktem T3, v opačném případě přejde do "čekacího taktu", ve kterém opět testuje signál READY. Východem z tohoto taktu je, že READY bude v čase tylové hrany impulzu $\Phi 2$ mít úroveň log 1.

Při činnosti čtení z paměti (obr. č. 11) je činnost synchronizace obdobná, až na to, že vložení taktu TW bude provedeno tehdy, když v kritickém čase taktu T2 bude cyklus VIDEO.

1 d. logika interfejsu klávesnice

- obr. č. 13 je realizována programovatelným paralelním interfejsovým obvodem MHB 8255A. Je možné ho považovat za interní (služební) interfejs, protože uživatel nemá hardwareovou přístupnost. Jeho úlohou je obsluhovat klávesnici, indikační diody LED a akustická návěští. Představuje určitou podporu programovým procedurám. Je vyveden k 20 pinovému internímu konektoru, který navazuje na klávesnicový konektor. V případě, že uživatel by po-

žadoval centrální desku použitelnou pro klasický alfanumerický počítačový systém, je možné k němu připojit (paralelně) displej.

Ovládání akustického návěstí je programově, prostřednictvím kanálu C obvodu MHB 8255A, který uvolňuje dvě frekvence z rozkladu obrazu na součtových hradlech NAND s otevřeným kolektorem. Jejich výstup přes tranzistor spíná elektromagnetický měnič umístěný na desce klávesnice.

1 e. Vstupně-výstupní strana mikropočítačové desky

- obr. č. 14 umožňuje rozšířit systém prostřednictvím interfejsových obvodů, kterých může být celkem 16. Z lokální sběrnice je vyvedena přes dva paralelně zapojené 30 pinové konektory FRB podmnožina sběrnice, která umožňuje připojit libovolný interfejsový obvod. Počítač PMD 85 má v jedné pozici připojení interfejsovou desku pro nejrozšířenější vstup/výstup a v druhé pozici se nachází ROM modul. Je nutno upozornit, že tato vyvedená sběrnice není oddělena ani výkonově zesílená, takže pokud by uživatel chtěl místo ROM modulu použít jinou vlastní konstrukci, musí s touto skutečností počítat. (uživatel má k dispozici tuto sběrnici přizpůsobenou k externímu používání, na interfejsové desce).

2. Klávesnice

Představuje samostatnou část mikropočítače (obr. č. 15), která je připojena přes 20 pinový konektor na základní počítačovou desku, a to konkrétně k obvodu MHB 8255 A. Je řešena jako matice spínačů s organizací 15 sloupců x 5 řádků. Sloupce jsou dekodovány prostřednictvím obvodu MH 74154 a ošetřeny diodami proti náhodnému zatlačení dvou tlačítek. Ovládání dekodéru je programovou exekutivou, která má za úlohu přečíst jednotlivé sloupce a současně zjistit aktivní řádek. Z těchto údajů je

zjištěn ASCII kód případně přivedený speciální kód. Kromě programovacích kláves zapojených v matici, obsahuje logika klávesnice ještě systémové klávesy - RESET, STOP, SHIFT.

Pro všeobecné použití - k indikaci požadovaných stavů jsou umístěny dvě diody LED, které se dají ovládat přímo linkami portu C služebního MHB 8255A. Jak bylo dříve uvedeno je zde umístěn elektromagnetický měnič. Ovládání těchto výstupů bude popsáno v části FIRMWARE.

Je třeba ještě podotknout, že aktivování tlačítka RES - tedy nulování (tzv. studený start) systému je podmíněn zatlačením tlačítka SHIFT, což zabráňuje náhodnému vynulování celého systému. Pohled na celkovou klávesnici s rozložením jednotlivých kláves je na obr.č. 16. Klávesnice obsahuje tři druhy typů kláves:

- standardní klávesy ASCII
- řídicí klávesy
- programové klávesy (software keys)

Jejich význam je následující:

- standardní klávesy ASCII přiřazují kód alfanumerického znaku. Některé klávesy mají přiřazeny dva kódy. Tento druhý kód se aktivuje stlačením klávesy SHIFT. Programová exekutiva zpracovává a přiřazuje kódy pouze pro velká písmena.

- řídicí klávesy

Mají specifický význam, protože jsou problémově orientované. Jsou to v podstatě klávesy, jejichž kód je použit pro vyvolání ediční exekutivy na dialogovém řádku. Jsou to tyto klávesy: WRK (WRITE KEY) - umožňuje zapisovat obsah dialogového řádku do bezprostředně zatlačeného tlačítka KEY (K0 - K11). Při zatlačení jiné klávesy tato funkce se automaticky zruší. RCL - (RECALL) opětovné přivolání posledního odeslaného obsahu dialogového řádku. Tento obsah se vypisuje prostřednictvím kurzoru.

INS - (INSERT) vkládání mezer na místo kurzoru

PTL - (PRINT ALL) nastavení příznaku pro výpis obsahu odeslaného dialogového řádku do obrazové části.

DEL - (DELET) mazání znaku určeného kurzorem.

CLR - (CLEAR) vymazání řádku od kurzoru vpravo (+ SHIFT),

CLR - vymazání řádku

- kurzor vlevo

- kurzor na začátek řádku

- kurzor vpravo

- rolování řádku vlevo

- rolování řádku vpravo

- kurzor na poslední znak v řádku

EOL - (END OF LINE) ukončení práce na obsahu dialogového řádku -

- ukončovací příkaz

C - D - zpráva systému o posledním jeho stavu. Text je vypisován do pomocného (návěstného) výstupního zobrazovacího registru.

- programové klávesy

Těchto kláves je celkově 12 a jsou označeny K0 - K11. Je možné je rozšířit na dvojnásobný počet s použitím klávesy SHIFT. Je možné na ně ukládat obsah jednoho dialogového řádku. Jeho zápis se provede za pomoci klávesy WRK a čtení jen samotným zatlačením příslušného KEY. Výpis je prostřednictvím kurzoru, což umožňuje efektivně pracovat při stereotypním používání jednotlivých výrazů.

3. ROM modul

obr. č. 17, svou strukturou patří mezi nevolatelné vnější paměti. Obsahují 16 pozic pro paměti ROM typu MHB 8707 (nebo 8608), což znamená 16 k byte nebo dále možnost osadit 32 k byte pomocí obvodů s organizací 2 k x 8. Adresace paměťového bloku

je realizována programovacími prostředky s podporou interfejsového obvodu MHB 8255A. Dekódování jednotlivých stránek paměti je realizována z dekodéru MH 74154, který je také ovládán programem. Vzhledem k tomu, že ROM modul napatří do oblasti operační paměti, se používá jako "knihovna" programových modulů. Paměti se aktivují pouze během přenosu dat - je to umožněno odpojováním pracovního napětí +12 V.

4. MODUL INTERFEJSŮ

Posláním modulu interfejsů je umožnit styk s okolím s periferními zařízeními. Obsahuje:

- dva paralelní programovatelné vstupy/výstupy (GPIO)
- jeden standardní styk IMS - 2 (HP-IB)
- jeden standardní sériový styk V24
- jeden styk pro komerční kazetový magnetofon
- jeden časovač s podporou hodin reálného času
- aplikační (všeobecný) vstup/výstup lokální sběrnice mikropočítače.

Připojení samotných interfejsových obvodů a jejich volba je zobrazena na obr. č. 18. Jak je z něho vidět, jde o standardní připojení ke sběrnici mikropočítače. Dekódování - výběr jednotlivých kanálů je realizováno obvodem MH 3205 z adresových linek A7, A6, A5, A4.

O jednotlivých výše uvedených interfejsích podrobněji:

- paralelní programovatelný vstup/výstup obr. č. 19

tento interfejs označený jako GPIO umožňuje pro dva samostatně pracující kanály práci s periferními zařízeními, které jsou orientovány pro paralelní styk (děrovač, snímač pásky, tiskárna apod.). Každý kanál je ošetřen - oddělen budičem MH 8287,

který může uživatel prostřednictvím vyvedeného řídicího signálu ovládat pro vstup případně výstup. V klidovém stavu je tento budič navolený (inicializovaný) na výstup. Připojením GND na tuto linku je umožněn vstup dat. Na konektor jsou vyvedeny i čtyři linky z interfejsového obvodu MHB 8295A - kanál C a tři invertory MH 7405, které může uživatel ze své strany použít pro úpravu řídicích linek kanálu (linky portu C).

- interfejs IMS - 2

obr. č. 20 tento interfejs je znám pod označením HP-IB. Je řešen výlučně programovacími prostředky, s podporou hardware, který představuje pouze výkonové přizpůsobení vstupně-výstupních linek. Programová exekutiva umožňuje práci v režimu pevně navoleného PMD 85 jako "CONTROLLER". V případě, že uživatel nepoužívá tento interfejs, je možné z něj získat dva paralelní vstupně-výstupní kanály s určitými omezenými vlastnostmi, které vyplývají z připojení portu C a B. Je třeba upozornit uživatele, že v tomto případě obdrží na výstupním konektoru negativní údaje.

- sériový styk V24

obr. č. 21. Sériový interfejs, který je realizován obvodem USART MHB 8254, je společný i pro styk s magnetofonem. Vstupní signál (data) i vzorkovací kmitočet je přepínán podle volby. Vstupně-výstupní linky jsou galvanicky odděleny optočleny. Vzorkovací kmitočet může být společný s MG, popřípadě je možné přepojit z interního časovače 8253 pomocí přepojky na desce. Tuto alternativu je možné využít pokud požadujeme přenosovou rychlost a strobování 16 popřípadě 64 násobné.

- interfejs pro kazetový magnetofon obr. č. 22

Je realizován s podporou USART 8251 a jednoduchým dekodé-

rem dat a hodin. Strategie záznamu je zvolena modulováním dat z obvodu USART a impulzy 1200 Hz, které jsou současně hodinami pro USART ($T \times O$). Generátorem tohoto kmitočtu je obvod MHB 9500, který zaručuje dostatečnou stabilitu kmitočtu. Takto nabitá datová směs je pro zpětné dekódování nejprve amplitudně upravená v obvodech MAA 748 a A301D tak, jak ukazuje obr. č. 23. Tento číslicový signál slouží jako vstupní signál pro dekódování dat a hodin pro USART. Z každé hrany signálu za pomoci obvodu 7486 se generují krátké impulzy, které spouštějí monostabilní obvod 74121 nastavený na $3/4$. Z výstupu je odebrán signál, který reprezentuje hodinové impulzy ($R \times C$) pro vzorkování dat v USARTu. Vstupní data ($R \times D$) jsou ještě zpracovávána klopným obvodem MH-7474, který má za úkol upravit data tak, jak šly z USARTu při záznamu.

Přenosová rychlost dat je dána jen vzorkovacím kmitočtem (1200 Hz) a dále úpravou časové konstanty monostabilního obvodu.

časovač

obr. č. 24. Je realizován obvodem 8253 a generátorem sekundových pulzů MHB 1116. Obvod 8253 obsahuje tři samostatné časovače softwareově přístupné. Jeden z nich (TIMER 0) je úplně k dispozici uživateli - přístupný přes aplikační konektor. Druhý (TIMER 1) má zabudovány vstupní hodiny $\Phi 2$ (TTL) a uživatel má k dispozici jen výstup časovače s jeho ovládáním. Třetí časovač (TIMER 2) má sekundové prvky na vstupu CLK a možnost vnějšího ovládání. Používá se jako hodiny reálného času.

- vstupně-výstupní aplikační konektor

obr. č. 25. Pro uživatele je vyvedena část lokální sběrnice mikropočítače, která umožňuje připojit různé interfejsové obvody. Všechny signály jsou ošetřeny pomocí obvodů 8286 a 7404.

Z napájecích zdrojů je vyvedeno jen + 5 V, které může uživatel využít max. do odběru 0,3 A. Zbylé linky konektoru FRB jsou obsazeny signály patřícími interface pro časovač.

III. OPERAČNÍ SYSTÉM

Mikroprocesor 8080 dovoluje svými 16ti adresovými linkami A0 až A15 adresovat pracovní paměť v rozsahu 0 až 64 k byte. V mikropočítači PMD 85 je tato pracovní paměť (operační paměť počítače), jak ukazuje obr. č. 27, rozdělená na čtyři paměťové oblasti - paměťové stránky po 16 k byte.

Pozn: Pro určení paměťového místa bude použito hexadecimální značení. První dvě paměťové stránky od adresy 0000 až po adresu 7FFF jsou osazeny pamětmi RWM tzn.; že je možné do těchto pamětí informace zapisovat a tyto vložené informace číst. Tyto dvě paměťové stránky jsou plně k dispozici uživateli pro vkládání vlastních programů např. Basic, Assembler, Editor, případně jiné uživatelské programy. Na další paměťové stránce, která začíná od adresy 8000, je umístěn rezidentní operační systém v rozsahu 4 k byte. Operační systém je uložen do paměti typu ROM, což znamená, že informace z paměti se mohou jen číst. Zbývající část této paměťové stránky od adresy 9000 až po adresu BFFF je neobsazená pamětmi a je ponechána pro rozšíření operačního systému.

Poslední stránka, která začíná od adresy C000 a končí na adrese FFFF je označena jako videostránka, ze které jsou informace pomocí videoprocessoru transformovány a přenášeny na TV obrazovku. Tato paměťová stránka je obsazena pamětmi RWM. Celá videostránka je rozdělena na skupiny po 64 bytech.

Z každé skupiny počínaje adresou 0000 se postupně transformuje prvních 48 byte, které jsou videoprocesorem ukládány na TV obrazovku jako řádky zdola nahoru. K ostatním 16ti byte-ům z každé skupiny nemá videoprocesor přístup a tyto jsou využity jako zápisníková paměť pro operační systém. Z transformované jednotky - byte je zobrazeno 6 bitů, a to zleva doprava, počínajíc od nejbližšího bitu D0. Bity D6 a D7 nejsou zobrazeny a jsou využity ve videoprocesoru pro změnu jasu, blikání nebo přenos barevné informace pro zobrazovanou šestici bitů z daného byte. Uspořádání transformace je znázorněno na obr. č. 28.

Videostránka je z funkčního hlediska rozdělena na dvě části a to:

a/ Část dialogového řádku. Tato paměťová část je transformována do dolní části TV obrazovky. Počet zobrazovaných znaků v tomto dialogovém řádku je 48, avšak délka dialogového řádku je 80 znaků, které se dají dynamicky posouvat pomocí edičních kláves a tedy je zobrazovat v rozsahu 48 znaků.

b/ Část pracovní. Tuto část využívá operační systém, kam odesílá k zobrazení svůj výstup informací. V případě alfanumerického zpracování informací je tento prostor orientován do formátu 48 znaků v řádku v 25 řádcích. Při zobrazování grafických výsledků je tento prostor rozdělen na 288 x 243 bodů. Pro program Interpolator je tento prostor snížen na 256 x 243 bodů.

Popis činnosti operačního systému

Start operačního systému

Mikroprocesor 8080 po jeho resetování vydává adresu 0000, na které očekává povel (instrukci) pro zahájení své činnosti. Obvodovým uspořádáním (negace adresy A 15) je dosaženo, že mikroprocesor čte počáteční instrukce uložené v paměti na adrese 8000

a po vykonání inicializačního programu se systém vrací do své normální činnosti.

Skladba operačního systému

Operační systém se skládá z těchto programových modulů:

MONITOR - programový modul = monitor
EDIT - Editor dialogového řádku
KLAV - programový modul pro spolupráci s klávesnicí
PRTOUT - programový modul pro výpis znaků ASCII
INPOL - Interpolátor
OSIO - programový modul pro vstup-výstup OS

1. MONITOR

Zabezpečuje jednoduché služby na úrovni strojového jazyka mikropočítače 8080. Obsahuje tyto povely:

SUB - modifikace paměti
MEM - výpis obsahu paměti do dialogového řádku
DUMP - výpis paměti do pracovní části obrazovky
JUMP - odstartování programu od zvolené adresy
MGSV - zápis programu na pásku kazetového magnetofonu
MGLD - zápis programu z pásky magnetofonu do počítače
MGEND - vyhledání konce zvoleného programu na magnetofonové pásce
JOB - načítání programu z programového modulu
BASIC-G - načítání jazyka BASIC - G z programového modulu

Poznámka:

V syntaxi povelu symbol $_$ znamená znak mezeru.

- SUB - modifikování paměti

Syntax: SUB $_$ adr. seznam hexa dat

přičemž: adresa představuje počáteční místo vkládání - adresu

zadanou v hexa. Při chybném zadání OS vypisuje hlášení:
"ERROR IN ADDRES".

- seznam je řetězec hexa dat, které se mají uložit od dané paměťové lokace. Mezi jednotlivými dvojicemi dat může být vložena i mezera. V případě, že se má vkládat řetězec znaků, musí být tento vložen mezi apostrof. V případě, že je chybně vložen hexa znak (0 - F) OS vypisuje chybové hlášení: "ERROR IN DATA".

Při správném přijetí příkazu se objeví příkaz SUB s následující adresou. Je možné vkládat minimálně jedna data nebo podle potřeby více dat až do konce dialogového řádku.

Příklad: SUB_L5000_L55_LAA_L0102_L GRAPHIC _L51

stlačení EOL

SUB_L5000

stlačení EOL

++ NO DATA ++

- MEM - zobrazení obsahu paměti od zvolené adresy po adresu + 16 do dialogového řádku.

Syntax: MEM_L adr.

příčemž: adresa představuje stejnou syntaxi jako u příkazu SUB.

Zobrazovaná data jsou s mezerou a nezačínají od začátku dialogového řádku. Tento vynechaný prostor umožňuje dopsat příkaz SUB s požadovanou adresou a tím data přesunout, případně editovat a potom zapsat do zvolené paměťové lokace.

- DUMP - výpis stránky paměti do pracovní části obrazovky.

Formát výpisu:

adresa

8 x data

ASCII znaky

Syntax: DUMP_{file} adr.

příčemž: adresa udává paměťovou lokaci, od které má výpis začít.

Po zobrazení výpisu se objeví v dialogovém řádku žádost (CONTINUE?) o pokračování výpisu. Pokud je požadován další výpis, stiskem klávesy EOL se vypíše další řádek.

Tento příkaz je možné výhodně použít při modifikování paměti s podporou příkazů MEM a SUB.

- JUMP - odstartování programu od zvolené adresy.

Syntax: JUMP_{file} adr.

Po vykonání tohoto příkazu se objeví v dialogovém řádku text "EXECUTIVE".

- MGSV - příkaz umožňující zápis dat na pásku kazetového magnetofonu.

Syntax: MGSV_{file}, adr. 1 - adr. 2, komentář

příčemž: file udává číslo (decimálně) záznamu a může být v rozsahu 00 - 63, adr. 1, počáteční adresa, adr. 2, koncová

adresa, komentář (nepovinně) osm znaků ASCII

Příklad: MGSV₀₁, 0000 - 10000, Basic

Po ukončení zápisu OS oznamuje hlášením do dialogového řádku "MG STOP".

Poznámka:

doporučuje se vést o příslušné kazetě písemný záznam o obsahu, aby se zaručila kontinuita číslování.

- MGLD - příkaz umožňující načítání dat ze zvoleného záznamu do operační paměti.

Syntax: MGLD_{file}

příčemž: file udává číslo (decimálně) záznamu, který se má vyhledávat a načítat do paměti.

Při hledání požadovaného záznamu se do dialogového řádku vypisuje hlavička předcházejícího záznamu s krátkým akustickým návěstím.

Příklad: 01/> TANK

; udává, že se jedná o záznam s pořadovým číslem 01 a patří pro BASIC.

Příklad: 01/D TANK

; udává, že se jedná o datový záznam patřící programu BASIC, tj.: že je přístupný pouze interpretováním programu BASIC.

Příklad: 01/? TANK

; udává, že se jedná o binární program, který je možné přečíst pouze z operačního systému.

Poznámka: jak je uvedeno v jednotlivých příkladech, může být záznam očíslován stejnými čísly, ale musí mít rozdílnou příslušnost.

Během čtení požadovaného záznamu se kontroluje výsledná suma a ta musí být shodná se sumou, která byla zaznamenána. V opačném případě vypisuje chybová hlášení: "FILLE ERROR".

- MGEND - příkaz umožňující vyhledat konec příslušného záznamu při současné kontrole zaznamenávaného kontrolního součtu.

Syntax: MGEND file

Příklad: MGEND 02

provádí obdobnou funkci jako příkaz MGLD až na to, že se data nenačítají do paměti. Tento příkaz se používá k verifikaci zapsaného záznamu, nebo pro přidání dalšího záznamu. Při zvoleném režimu vyhledávání záznamu má uživatel možnost přerušit činnost dialogu mezi počítačem a magnetofonem zatlačením tlačítka STOP. Jako odpověď systému se mu vypíše hlá-

ení do dialogového řádku "FILLE ERROR". Přerušit činnost hledání záznamu není možné v případě načítání dat do paměti.

- JOB - příkaz umožňující přenos údajů programu z ROM modulu PMD 85 do operační paměti. Po jeho přenosu se provede rovněž jeho odstartování od první paměťové lokace.

Syntax: JOBUkód

příčemž: kód znamená 12ti místné hexa číslo, které se v podstatě reprezentuje následovně:

odkud	kolik	komu
ROM adresa	počet	RAM adresa

Například: pro zavedení interpreteru BASIC - G je možné použít následující příkaz:

JOB ~~0000~~0250~~000000~~

BASIC - G - příkaz umožňující práci s vyšším programovacím jazykem BASIC.

Syntax: BASIC G

Po odeslání příkazu se provede přesun interpreteru z modulu paměti ROM do operační paměti a jeho odstartování. Tím bylo provedeno opuštění MONITORu a přechod na zvolený programový modul.

Kromě těchto standardních povelů může uživatel s výhodou využít při sestavování vlastních programů celou řadu podprogramů z programového modulu MONITOR. Předpokládá se však znalost systému 8080 a jeho programování. Toto platí také při využití dalších programových modulů a jejich podprogramů.

Podprogramy modulu MONITOR

- HEX - podprogram pro převod čísla ASCII do HEX. V případě chyby, že se nejedná o znak z množiny hexa znaků (0 - F) se nastavuje CARRY bit.

Vstup: akumulátor

Výstup: akumulátor a příznakový bit CARRY

Volání: 80E0H

Ničí registry: PSW

- PAIR IN - převod 2 x ASCII do HEXA

Vstup: ukazatel HL

Výstup: akumulátor a příznakový bit CARRY

Volání: 80F7H

Ničí registr: PSW, HL, B

- ADRIN - převod 4 x ASCII do 2 x HEXA

Vstup: ukazatel HL

Výstup: registrový pár DE, příznakový bit CARRY

Volání: 8109H

Ničí registr: všechny

- PREVOD 1 - převod hexa na dva ASCII při jejich současném zobrazení do pracovní části obrazovky. Využívá podprogram PRTOUT.

Vstup: znak HEXA v akumulátoru

Výstup: rutina PRTOUT (8500)

Ničí registr: B, PSW

Volání: 8125H

- PREVOD 2 - převod HEXA na dva ASCII a jejich uložení do paměti na místo určené ukazatelem v HL. (první znak)

Vstup: znak HEXA v akumulátoru, ukazatel HL

Výstup: paměťové místo určené v HL

Volání: 813BH

Ničí registr: B, PSW

- BIN/BCD - převod binárního čísla (hexa) na BCD v rozsahu
(0 - 63)

Volání: 8E73H

Vstup, Výstup: akumulátor

Ničí registry: HL, B, PSW

- TRANSFER - přenesení obsahu z ROM modulu do operační paměti.

Argumenty se ukládají za volací rutinou v pořadí:
počáteční adresa ROM, délka bloku, ukládací adre-
sa RAM.

Volání: 8C00H

Ničí registry: všechny

Příklad: CALL TRANSFER

CD 00 8C

DW ADR 1

00 00

DW BATCH

00 25

DW ADR 2

00 00

- STOP - podprogram zajišťuje zatlačení klávesy STOP. V případě,
že k tomu došlo, akumulátor nabude hodnoty 03.

Volání: 8C74H

Vstup: klávesa STOP

Výstup: PSW

Ničí registry: PSW

- BREAD - podprogram sloužící jako podpora pro grafický BASIC a
umožňuje načítat hodnotu byte z pracovní části obra-
zovky z místa, na které ukazuje kurzor.

Volání: 8473H

Vstup: ukazatel místa (kursor) 017A4

Výstupy: akumulátor

- RPOINT - podprogram sloužící jako podpora k interpolátoru (BASICu) a umožňuje ze zadaných souřadnic bodu zjistit jeho stav (0 nebo 1). Využívá podprogram POINT a s ním související jeho vstupní argumenty.

Volání: 85E64

Vstup: argumenty podprogramu POINT

Výstupy: akumulátor nabude hodnoty 00 nebo 01

Ničí registry: všechny

- MONIT - podprogram zabezpečuje vstup do monitoru, nastavuje prompt znak "?" a vypisuje text "OS READY" do dialogového řádku.

Volání: 8C404

2. EDIT

Tento programový modul zabezpečuje interaktivní činnost s obsahem dialogového řádku. Při přijetí vstupního argumentu se rozhodne, zda se jedná o alfanumerický nebo řídicí znak. V případě alfanumerického znaku se provede jeho okamžité rozkreslení nad pozici kursoru. Z ostatních přijatých znaků se provede třídění a vykoná se příslušný podprogram. Jedná se o tyto řídicí znaky:

→ kurzor vpravo
← kurzor vlevo
→ řádek vpravo
← řádek vlevo
CLR řádek vymaž

↑ CLR od kurzoru do konce řádku vymaž
↑ PTL zobraz do pracovní oblasti celý řádek
DEL vymaž znak v místě kurzoru
INS vlož mezeru v místě kurzoru
↖ kurzor na začátek řádku
ENDL kurzor na poslední znak v řádku
WRK bude se zapisovat do programových kláves (tzv. soft-keys)
RCL přivolání odeslaného řádku
EOL ukončení práce nad obsahem dialogového řádku
C-D přivolání návěstí od OS

volání: adr. 8800H

vstupní argument: znak na adresu C134H

Kromě těchto edičních podprogramů jsou zde podprogramy, které mohou být uživateli k dispozici. Jsou to:

BELL - podprogram zabezpečující akustický výstup

Volání: adr. 88A3H

Ničí registry: PSW, DE, HL

V případě vlastního komponování akustického výstupu je možné nastavit ukazatel tabulky do registru HL a volání uskutečnit s adresou 88A6H.

Složení tabulky:

TÓN	WAIT	TÓN	WAIT	MARKA
-----	------	-----	------	-------

příčemž:

- TÓN - může nabývat hodnoty 00, 01, 02, 03
- WAIT - hodnotu 00 - FF
- MARKA - hodnota FF ukončuje činnost a znamená výstup z rutiny BELL.

Příklad:	LXI, H, TAB	6000	21 06 06
	JMP BELL+1		C3 A6 88
TAB:	DB TON 1		01
	DB WAIT 1		50
	DB TON 2		02
	DB WAIT 2		80
	DB MARKA		FF

Příklad znázorňuje vytvoření podprogramu s volacím adresovým vektorem 6000 s akustickým výstupem daným hodnotami uloženými na TAB (adresa 6006).

ZOBR BUF - podprogram provede zobrazení řetězce uloženého v paměti, jehož začátek je dán ukazatelem v registrech HL, do dialogového řádku.

Volání: adr. 8858H

Ničí registr: všechny

Vstupní argument: adresa začátku řetězce do HL.

PRTTEXT - vytištění řetězce ASCII do dialogového řádku s možností akustického návěstí.

Volání: adr. 8AB0H + BELL

adr. 8ASC4 bez BELL

Ničí registry: všechny

Vstupní argument: ukazatel začátku řetězce uložený v zápisníku na adr. C074 a C075.

Poznámka: řetězec ASCII musí být ukončen znakem 0DH!

3. KLAV

Zabezpečuje přiřazení kódu ASCII znaků a řídicích znaků z klávesnice, které jsou umístěny v tabulce od adresy 8400 tak, že každá pátá obsahuje příslušnou značku určující kód řádku.

Pokud je nalezen příslušný řádek (snačku), potom se připočítá pozice klávesy v řádku (0 - 14) a tím je určen příslušný kód znaku. Ten je možno převzít přímo z akumulátoru nebo ze zápisníku adresy C134. Adresový vektor programovacího modulu je 84A1 a uchovává všechny registry až na PSW.

4. PRTOUT

Organizuje "rozkreslení" znaku ASCII do videostránky paměti RWM na místo, které udává argument kurzoru.

- vstupní argumenty:

- akumulátor .. znak výpisu

- údaje ze zápisníku - C03EH }
 C03FH } pozice výpisu (kurzor)

- C03A kód barvy

- C03CH }
 C03DH } tabulka znaků

Tyto údaje v zápisníku jsou inicializovány následovně:

C03E 00 C03A --00 (80, C0, 40)

C03F FB

C03C 00

C03D 85

Volání: adr. 8500H

Ničí registr: PSW

Tento programový modul obsahuje výkonný podprogram, který vyhledá z tabulky znaků příslušný kód a provede jeho rozkreslení do nastavené části paměti. Je volen přes adresu 8584H a používá výše uvedené vstupní argumenty.

Na obr. č. 29 je uvedena tabulka znaků, ze které je zřejmé, jak je možno modifikovat libovolné typy případně symboly v rastru 6 x 8 bodů. Je třeba zdůraznit, že hodnota ukazatele tabulky se

nastavuje při rozkreslení znaků v rastru 5 x 7 zmenšená o FF (hexa) a při plném vykreslování v rastru 6 x 8 bodů zmenšená o FE (hexa). Je to z důvodu rychlejšího přenosu při alfanumerickém výstupu.

Vraťme se ještě k exekutivě PRTOUT. Je třeba dodat, že jako vstupní znak kromě standardních alfanumerických znaků (i malá písmena) ASCII mohou být přijaty znaky:

- LF (0A hexa) - prázdný znak
- CR (0D hexa) - návrat na nový řádek + posun
- FS (1C hexa) - nulování obrazovky

Ostatní předcházející znaky nejsou respektovány a umožňují přímý výstup z exekutivy tj. pokud bychom dali vytisknout 80 znaků v řádku, tak se bude akceptovat pouze 48 a dále se očekává znak na nový řádek s posunem (CR).

5. INPOL

Tento programový modul zabezpečuje vytvoření lineárního spojení (interpolace) mezi dvěma libovolně zadanými body v pracovní části obrazovky (videostránky). Z důvodu větší rychlosti byla šířka pracovní části snížena na 256 bodů, což představuje jeden byte a samozřejmě všechny aritmetické operace s osmibitovými čísly.

Interpolátor požaduje následující vstupní argumenty, které jsou uloženy v zápisníku OS:

- C170 - hodnota X posledního bodu
- C172 - hodnota Y posledního bodu
- C173 - hodnota X žádaného bodu
- C174 - hodnota Y žádaného bodu

Volání: adr. 8CD0H

Ničí registry: všechny

Samotné zobrazení bodu zabezpečuje podprogram POINT, který musí vypočítat vzhledem k strojové nule (adresa C000 videostránky) příslušnou adresu a v samotném byte pozici bitu.

Volání POINT: adr. 8C7DH

Vstupní argumenty: C070H souřadnice X

C072H souřadnice Y

C03AH barevný kód

C1FAH mód kódu

příčemž "mód kódu" může nabývat pouze tyto hodnoty (hexa)

A8H..... negace bodu

AFH..... nulování bodu

B0H..... nastavení bodu

Ničí registr: PSW

Návrat z programového modulu INPOL nastane při ztotožnění údaje souřadnic bodu posledního s žádaným.

6. OSIO

Pro vzájemnou spolupráci operačního systému s uživatelským programem je nutná jejich správná korespondence. Na obr. č. 30 je znázorněno jeho umístění v OS. Volání podprogramu ENTER (adresa 8BEE) umožní odložení vrcholu zásobníku, návratové adresy a nastaví vlastní vrchol zásobníku na hodnotu 7FFF.

Výstup z podprogramu a pokračování v interpretování uživatelského programu se provede stlačením klávesy EOL.

Na základě toho se přesune obsah dialogového řádku do lokace zadané ukazatelem C078 a daný řetězec je ukončen znakem CR (0D). Dále se celý řádek uloží do oblasti vyhrazené pro programové klíče, který je možné zpět přivolat klávesou RCL. Potom se nastaví vrchol zásobníku, který byl před vstupem do

podprogramu a pokračuje se v interpretování přerušného programu.

- Programové klíče

Množina kláves K0 - K11 umožňuje uživateli uložit libovolný řetězec znaků do vyhrazené části paměti. Řetězec přijme znak klávesy (K0 - K11) jako identifikátor a v případě, že paměťová oblast je prázdná (nulová) je do ní uložen. Pokud se už v této části nachází řetězec se stejným identifikátorem, dojde k jeho zrušení a kompresi stávajících řetězců a až potom je na jejich konec přiřazen daný řetězec. Žádost, že se bude zapisovat řetězec, je systému ohlášen stlačením klávesy WRK, a až poté je třeba stisknout klávesu programového klíče. Když pracovní část určená pro zápis je nedostatečná, systém ohlašuje do dialogového řádku text "MEMORY OVERFLOW". Vymazání a tedy k zmenšení obsazenosti prostoru je možné provést zapisováním prázdného řádku. Při vypisování vloženého řetězce se jeho obsah vypisuje kurzorovým způsobem do dialogového řádku. V případě, že se vyvolá programový klíč, který není obsazený, systém ohlašuje zprávu "NO KEY".

Poznámka:

Obsah dialogového řádku ukončený klávesou EOL se nejprve přenesení do programovacího modulu, a potom se zapisuje do pracovní oblasti s možností jeho zpětného vyvolání. Taktéž pokud by byla tato oblast přeplněna, systém neumožní odeslání obsahu řádku do programového modulu, což potvrzuje hlášením "MEMORY OVERFLOW".

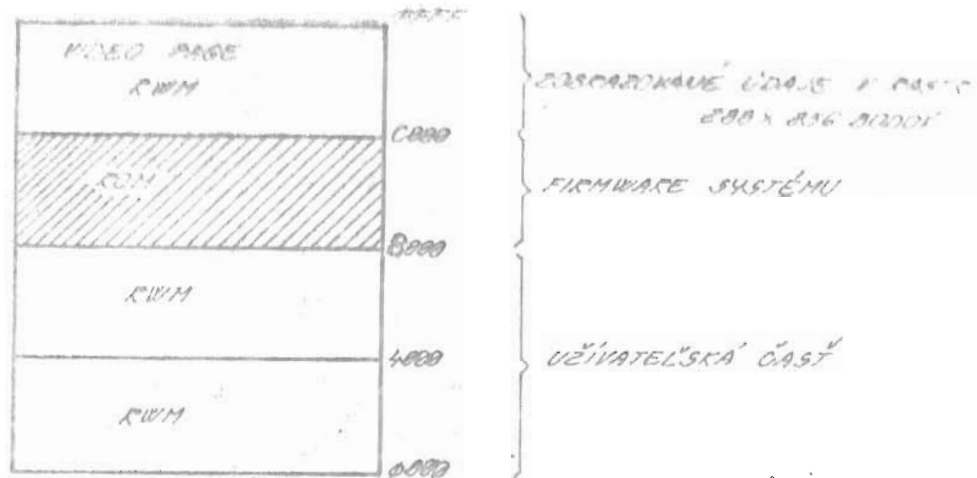
Modifikace systému

V této části jsou shrnuty argumenty, kterými můžeme přemodifikovat systém. Každou změnu je nutno zvážit, protože může dojít ke "zborcení systému". Ukazatele popřípadě jejich atributy jsou uvedeny i v decimálních formách, aby bylo možné je vyvo-

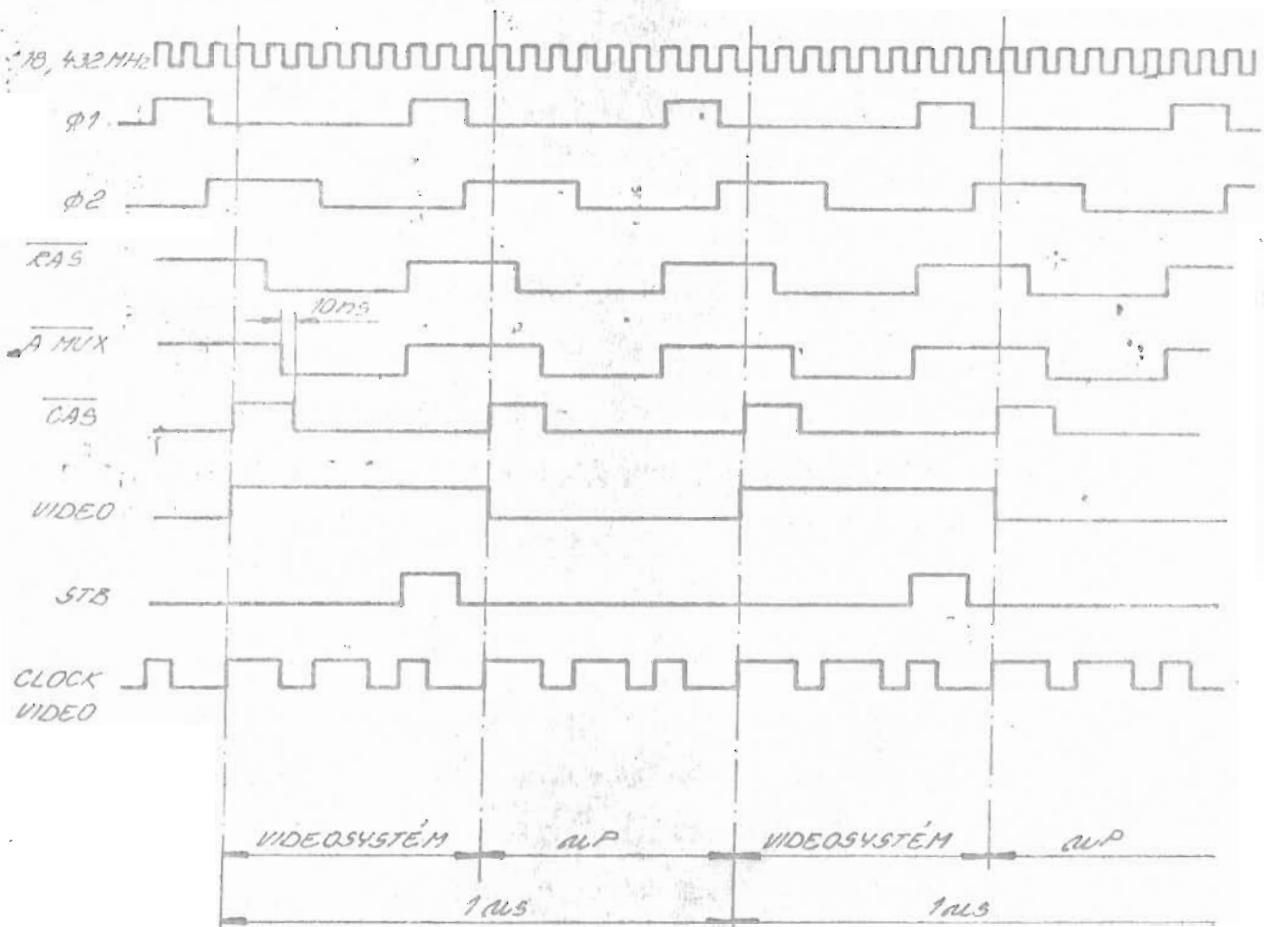
lávat prostřednictvím příkazů POKE, PEEK, USR, OUT, INT.

V některých případech hlavně při interpolaci, kde se vyžaduje velká rychlost interpretování programu, tato modifikace je uskutečněna přepisem typu instrukce v části programu, který je uložen v RAM.

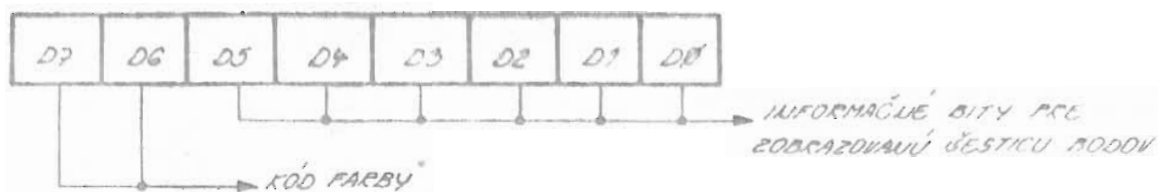




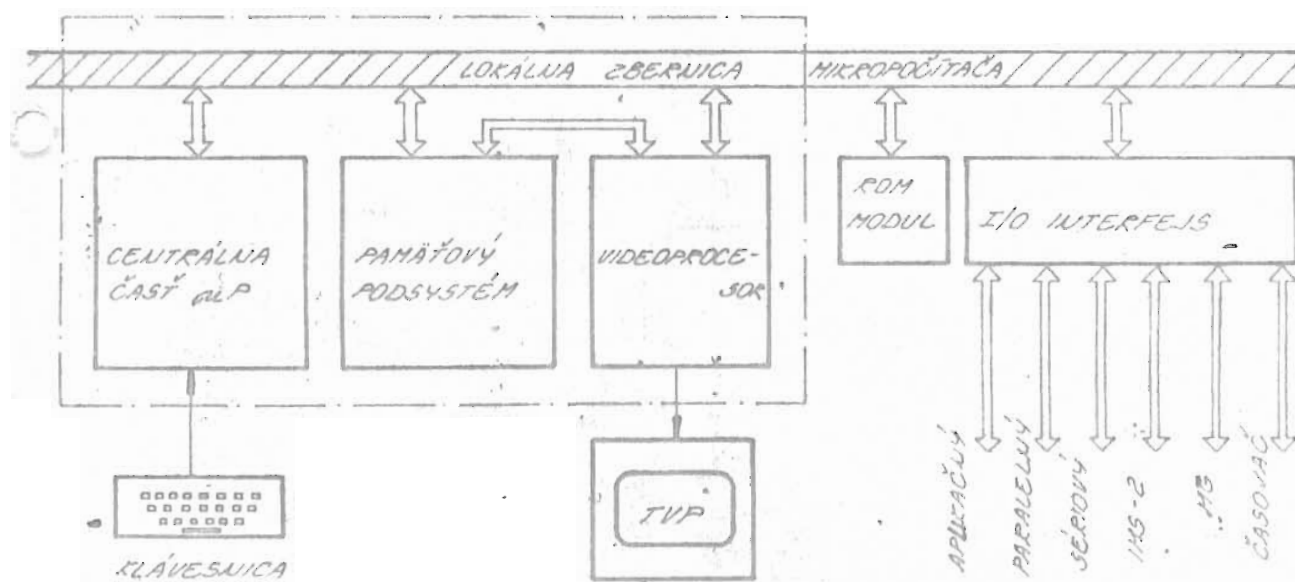
obr. č. 1 MAPA PAMĚTI



obr. č. 2 ZÁKLADNÍ ČASOVÁ SPOLUPRÁCA VIDEO SYSTÉMU A MIKROPOČÍTAČE



obr. č. 3 FORMÁT VIDEOÚDAJA

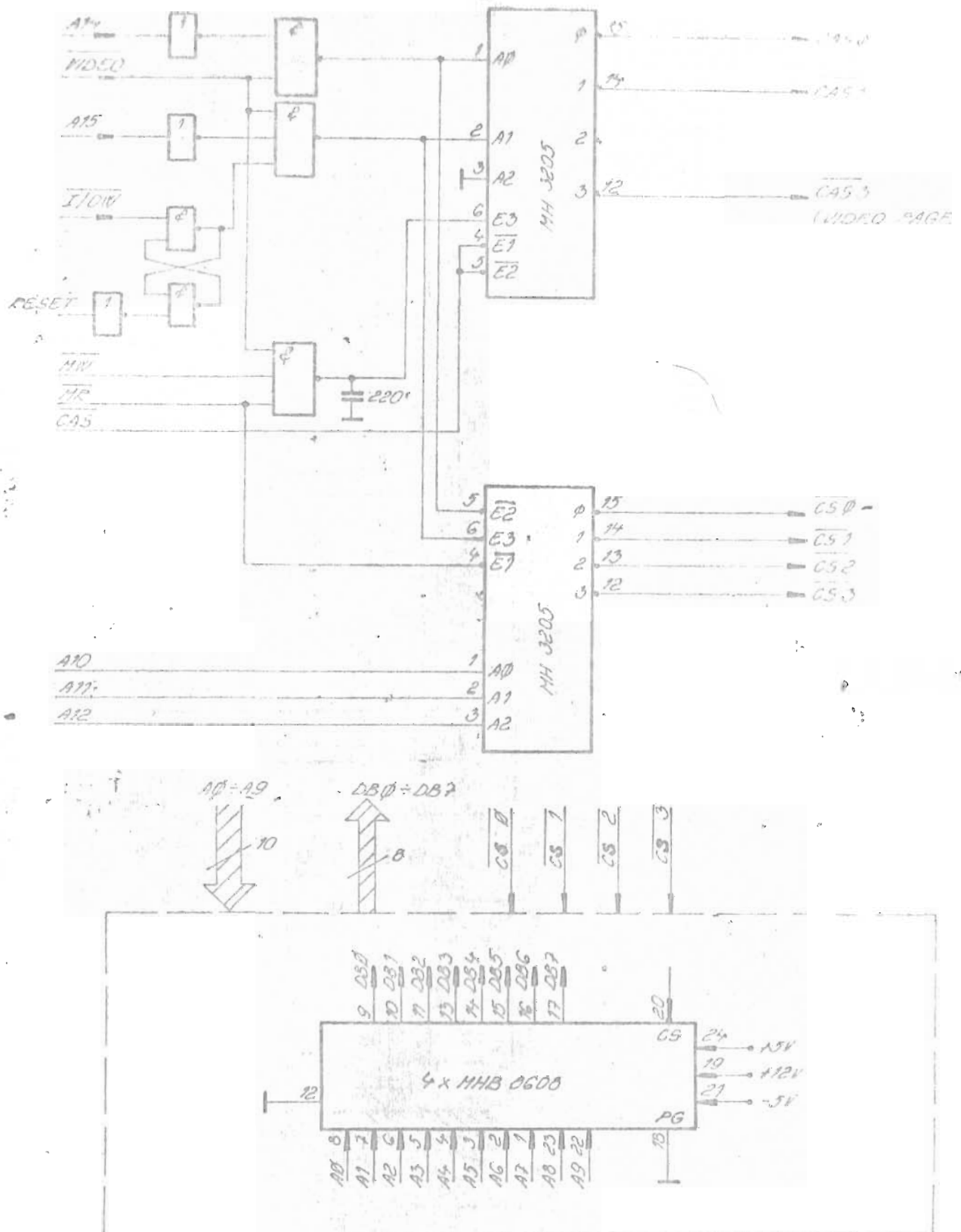


obr. č. 4 KONCEPCIA SYSTÉMU PHD-85

READY	HW	MR	VIDEO
0	0	1	0
1	1	0	0
1	1	1	0
1	0	1	1
0	1	0	1
1	1	1	1

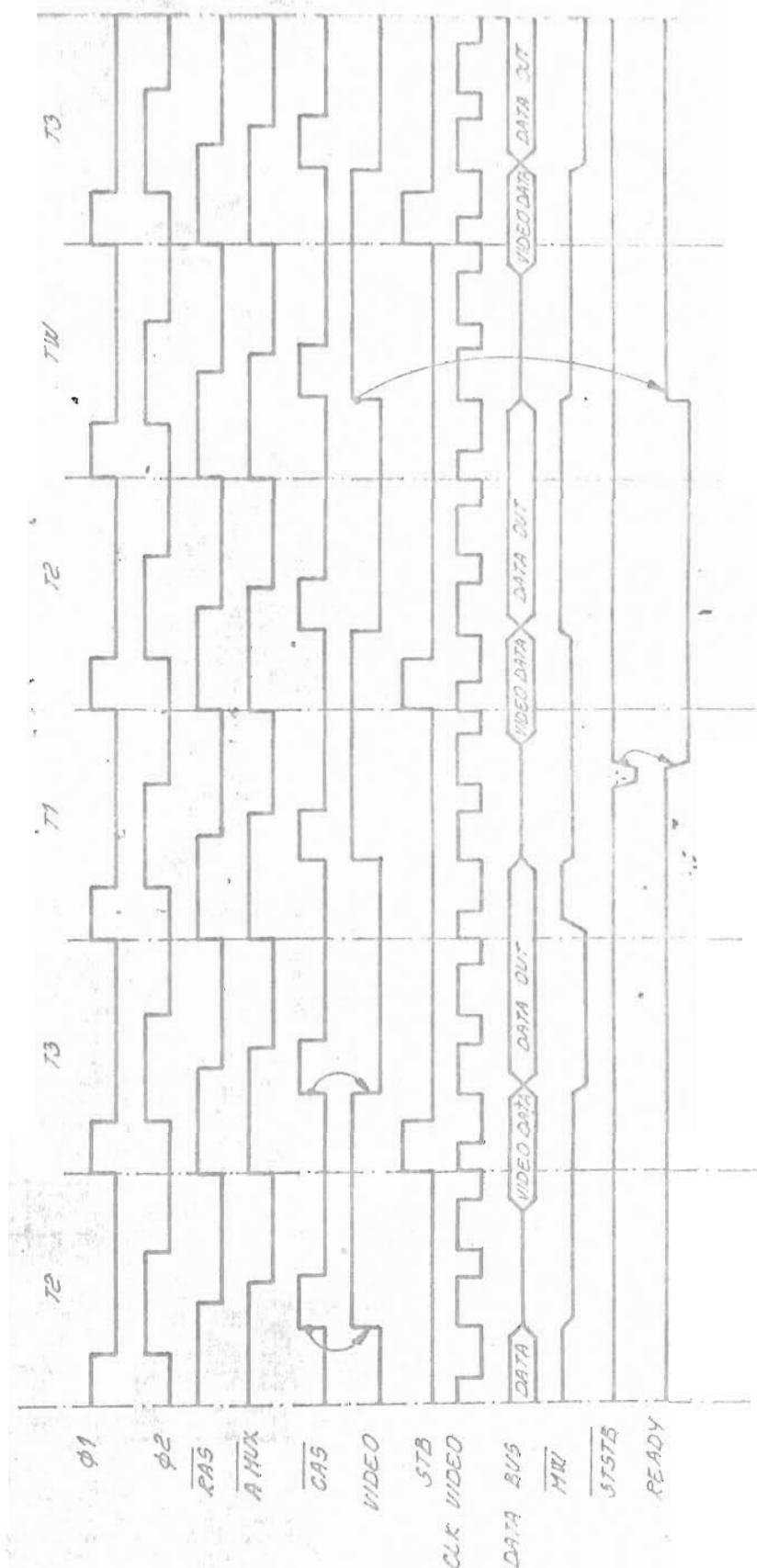
obr. č. 6 TABUĽKA URČUJÚCA TVORBU SIGNÁLU READY



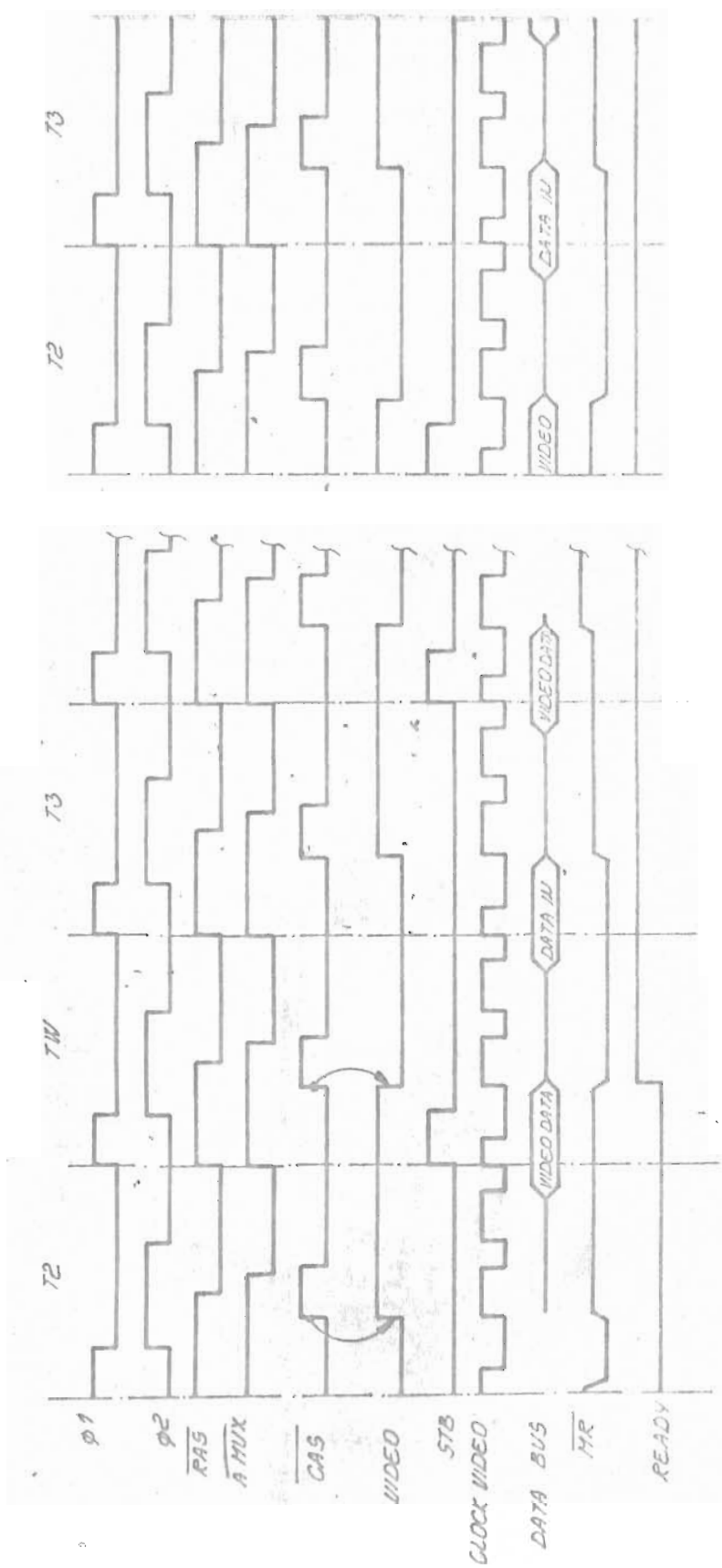


obr. č. 8 SCHEMA ZAPOJENIA PAMÄTOVÉHO PODSYSTÉMU IBM

OBR. Č. 10 SPOLUPRÁCA VIDEOSYSTÉMU S RLP POUŽÍV. ZÁPISU DO RAMĚTI



Obr. č. 17 SPOLUPRÁCA VIDEOSYSTÉMU S ADP PRI ČÍTANÍ 2 PAMATE R2W9



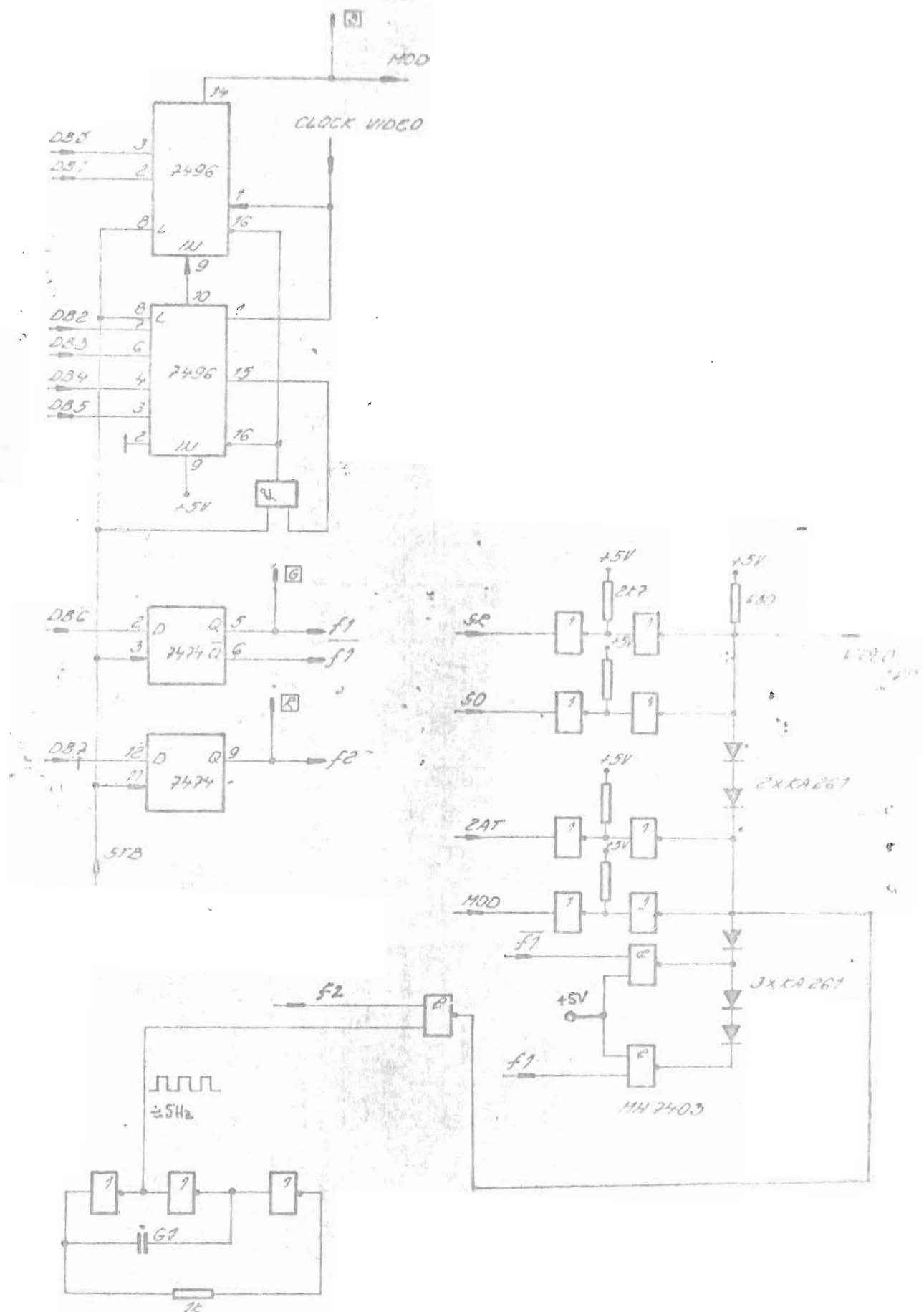
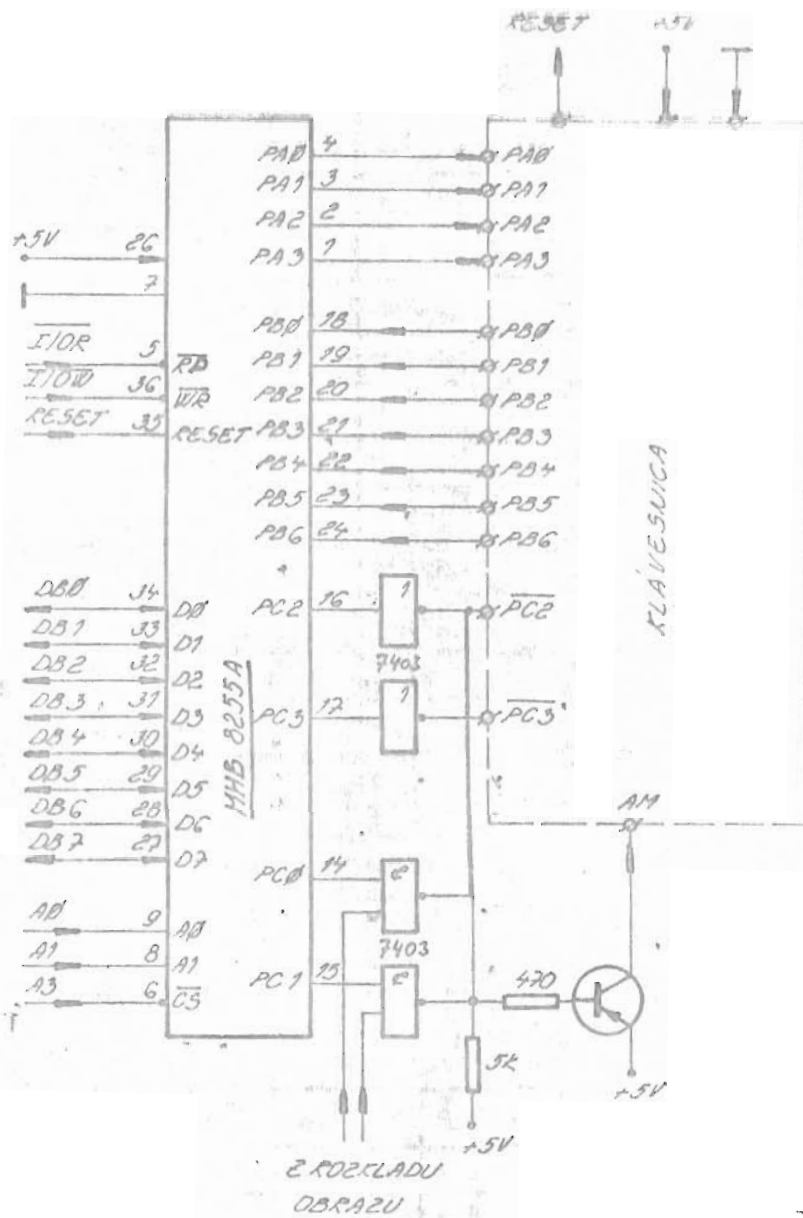


СХЕМА ЗАПОЈЕНІА СЕРІОПАРЛЕЛНОГО ВІСТУПНОГО РЕГІСТРА ДАТОВИ



MOD. 8255A	KANÁL	HEX
	PA	F4
	PB	F5
	PC	F6
	CM/R	F7

obr. č. 13 SCHEMA ZAPOJENIA INTERFEJSU KLAVESNICE

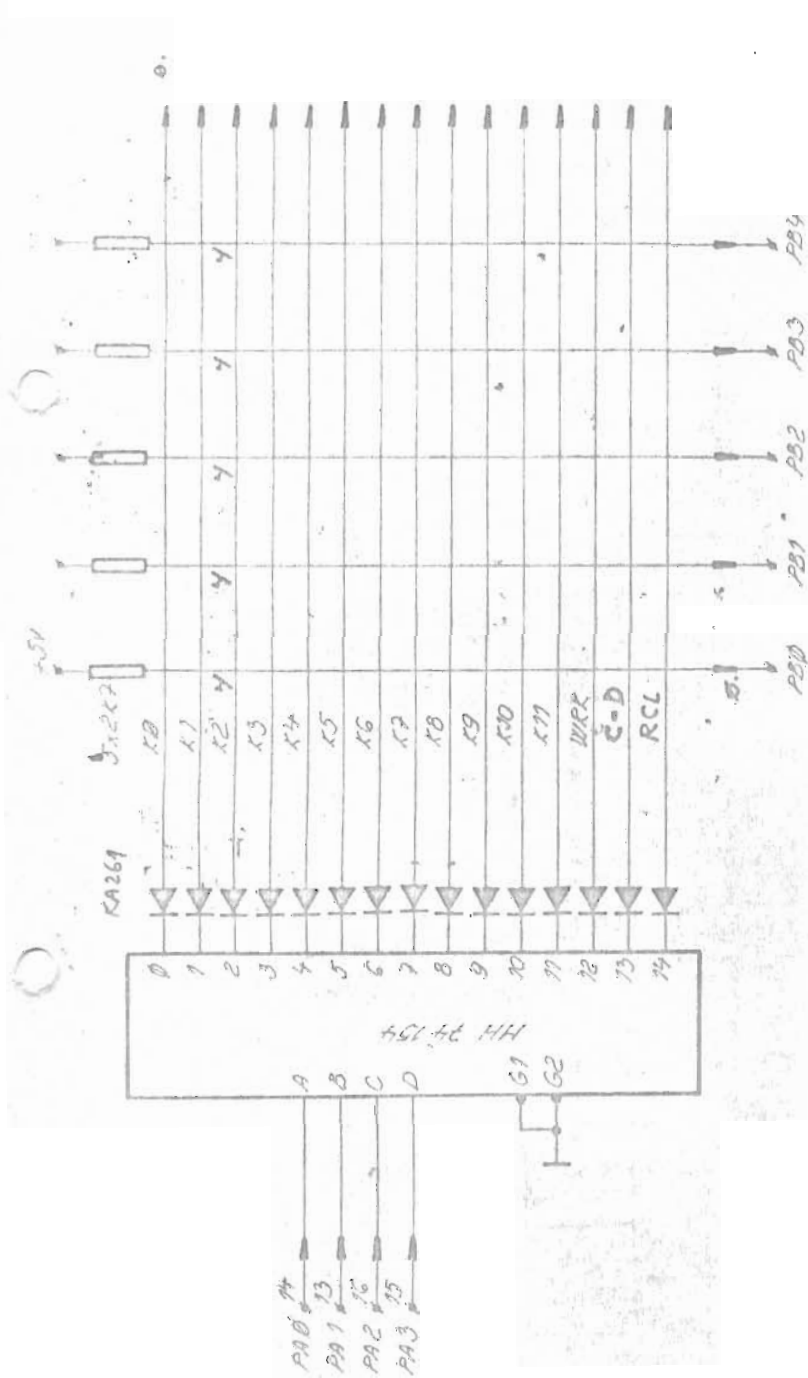
COMBINO ZOBRAZKA

SR	2	1	D7
INT	4	3	D6
P2 TTL	6	5	D5
A2	8	7	D4
A3	10	9	D3
A4	12	11	D2
A6	14	13	D1
A7	16	15	D0
A1	18	17	RESET
A0	20	19	I/O WR
I/O RD	22	21	A5
+12V	24	23	+12V
-5V	26	25	-5V
+5V	28	27	+5V
GND	30	29	GND

KLÁVESNICA

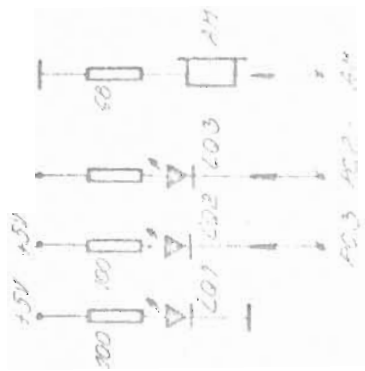
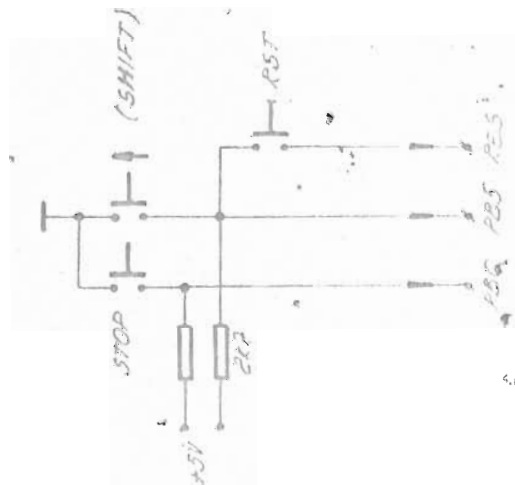
PC2 - LED 3	2	1	AM - AKUSTICKÝ MENIČ
PC3 - LED 2	4	3	RST - RESET
PB0 - RIADOK MATICE	6	5	PB6 - STOP
PB1 - RIADOK MATICE	8	7	PB5 - SHIFT
PB2 - RIADOK MATICE	10	9	PB4 - RIADOK MATICE
	12	11	PB3 - RIADOK MATICE
PA0 - STLPEC MATICE	14	13	PA1 - STLPEC MATICE
PA2 - STLPEC MATICE	16	15	PA3 - STLPEC MATICE
+5V	18	17	+5V
GND	20	19	GND

ОБТ. 8.15 SCHEMA ZAPOJENIA KLAVESNICE



STUPE

RIADKY



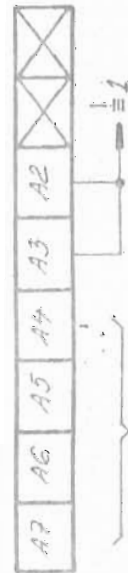
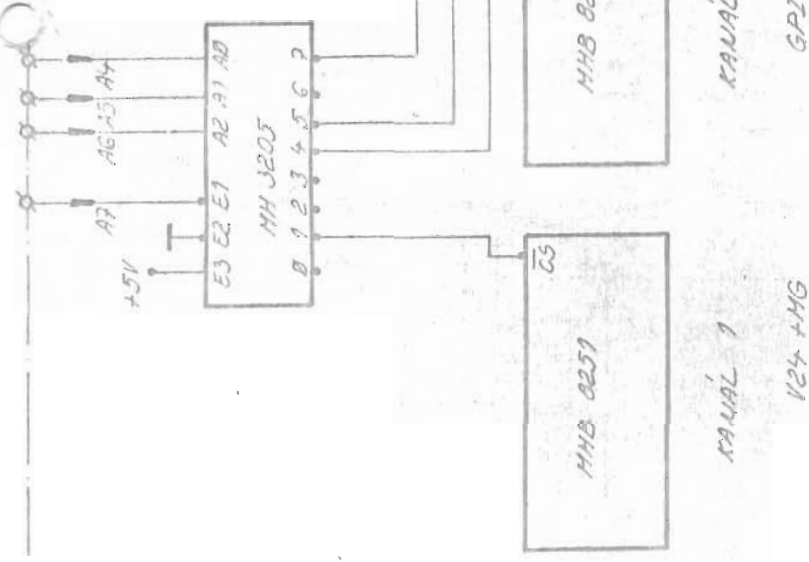
STÍPEČ

RIADOK

K0	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	WRK	C-D	RCL	RST
! 1	" 2	# 3	\$ 4	% 5	& 6	' 7	(8) 9	= 0	_		PTL INS	DEL	CLR	
Q	W	E	R	T	Z	U	I	O	P	Q	W	↩	↩	↩	↩
A	S	D	F	G	H	J	K	L	+	*	:	↩	↩	ENDL	↩
↑	Y	X	C	V		B	N	M	<	>	?	↑	STOP	EOL	

SPACE

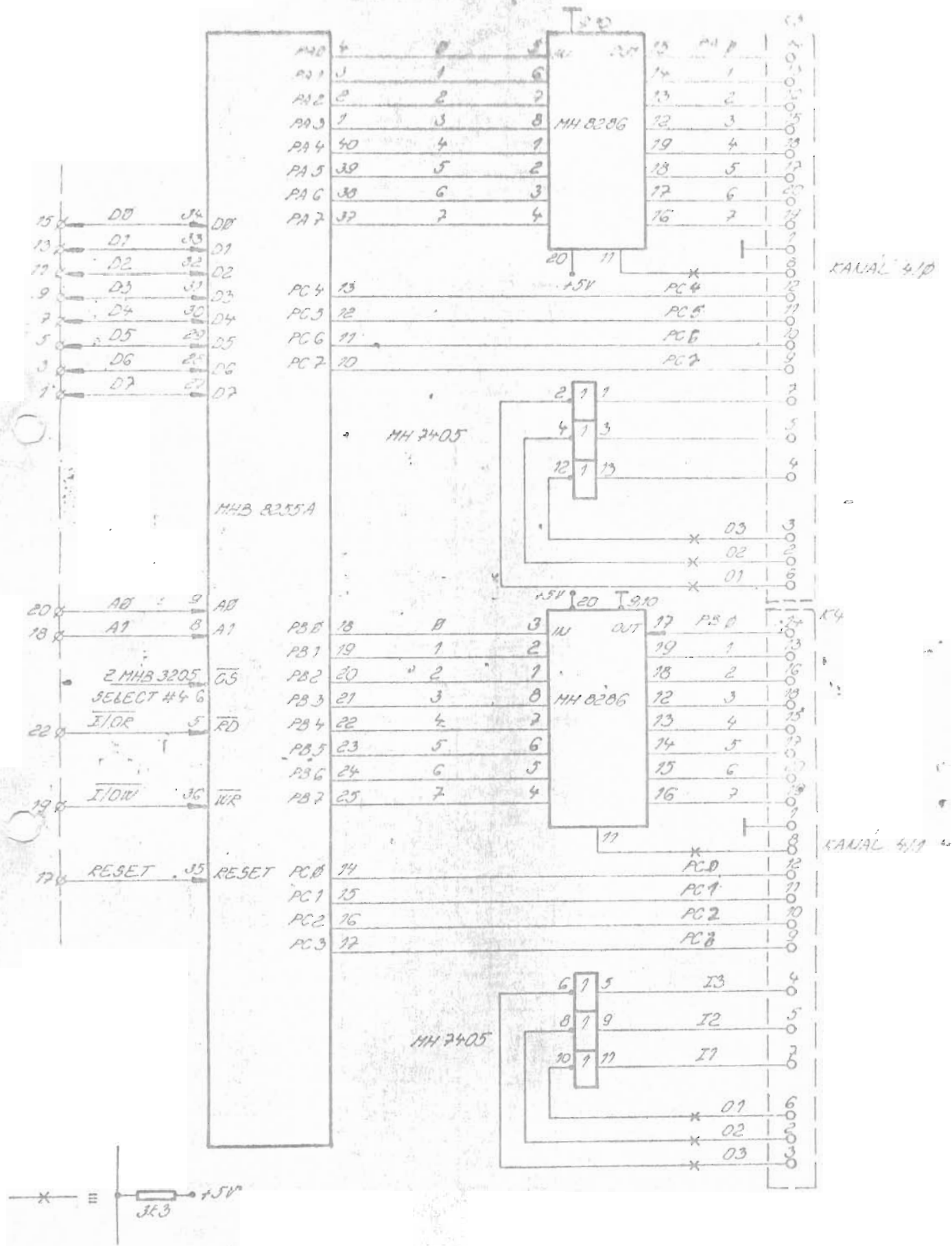
obr. č. 16 POHLAD NA KLAVESNICU



VÝBER INTERFEJSOVÝCH OBVODOV

	1	2	3	4	5	6	7	8
1	x	x	x	x	x	x	x	x
2	x	x	x	x	x	x	x	x
3	x	x	x	x	x	x	x	x
4	x	x	x	x	x	x	x	x
5	x	x	x	x	x	x	x	x
6	x	x	x	x	x	x	x	x
7	x	x	x	x	x	x	x	x
8	x	x	x	x	x	x	x	x

EXTERNÍ INTERFEJSY
INTERNÍ INTERFEJSY
SLUŽEBNÝ PPI 8255A
ROM MODUL



obr. č. 19 SCHEMA ZAPOJENIA PARALELNÉHO INTERFEJSU (GPIO)

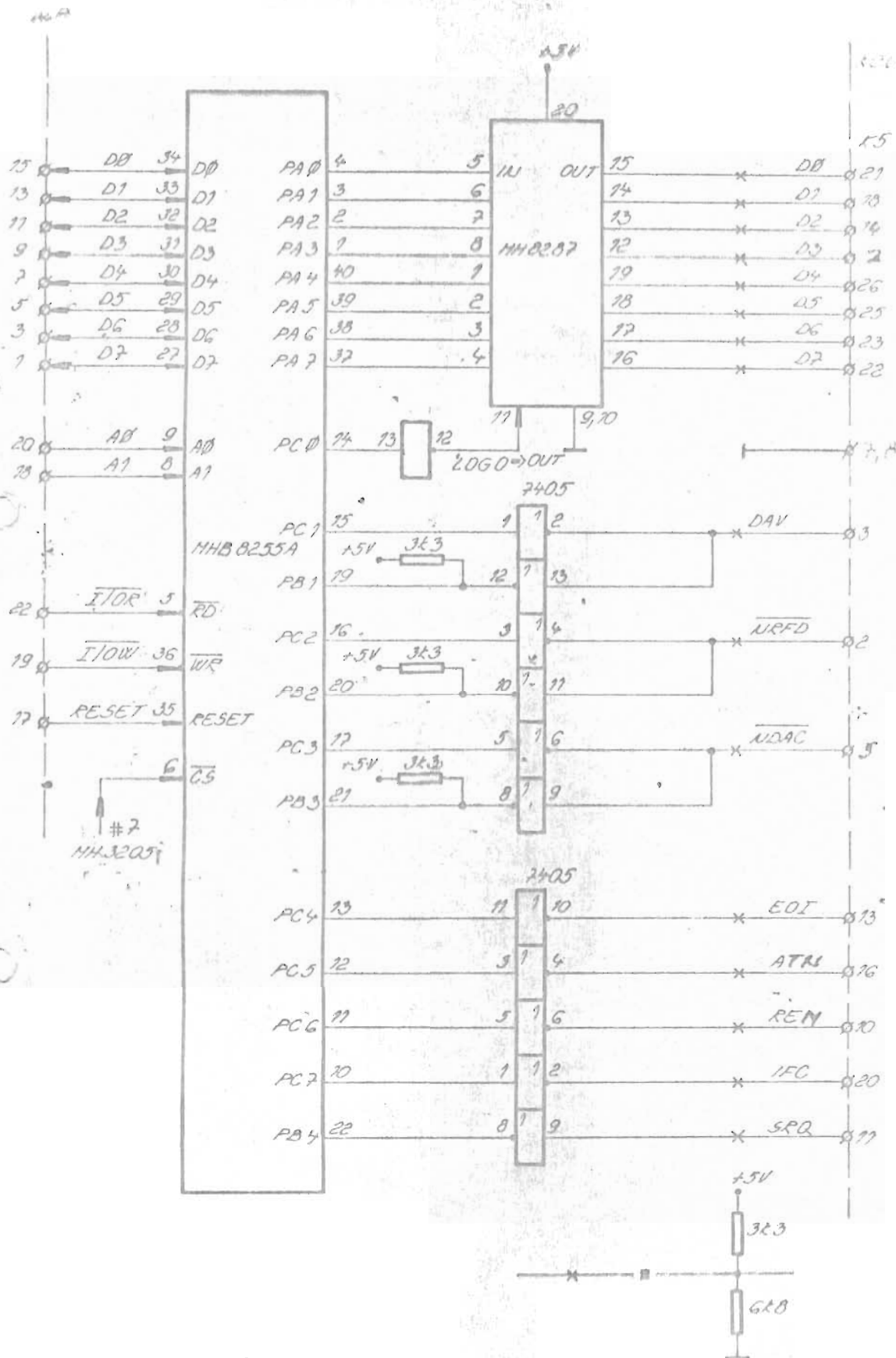
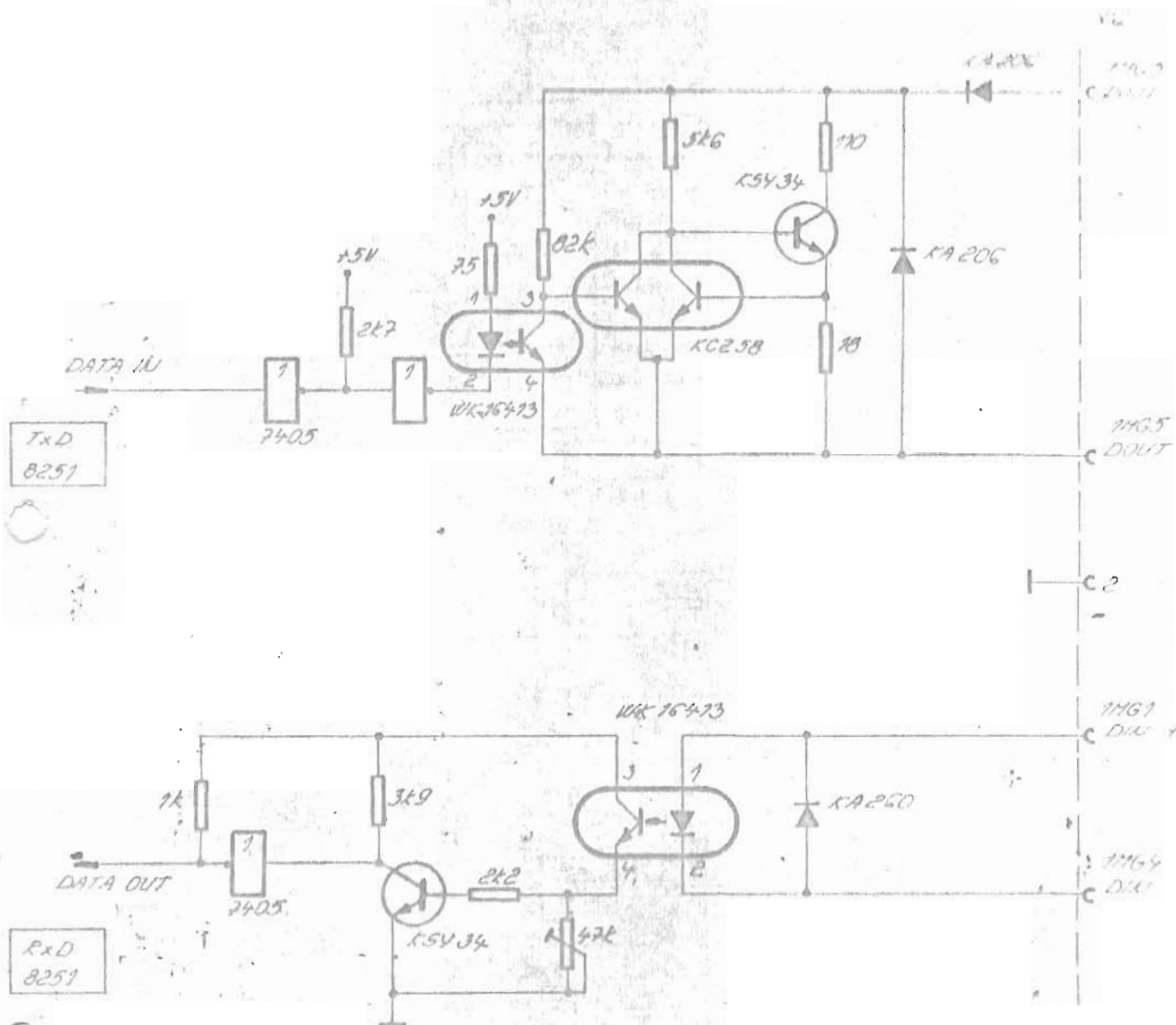
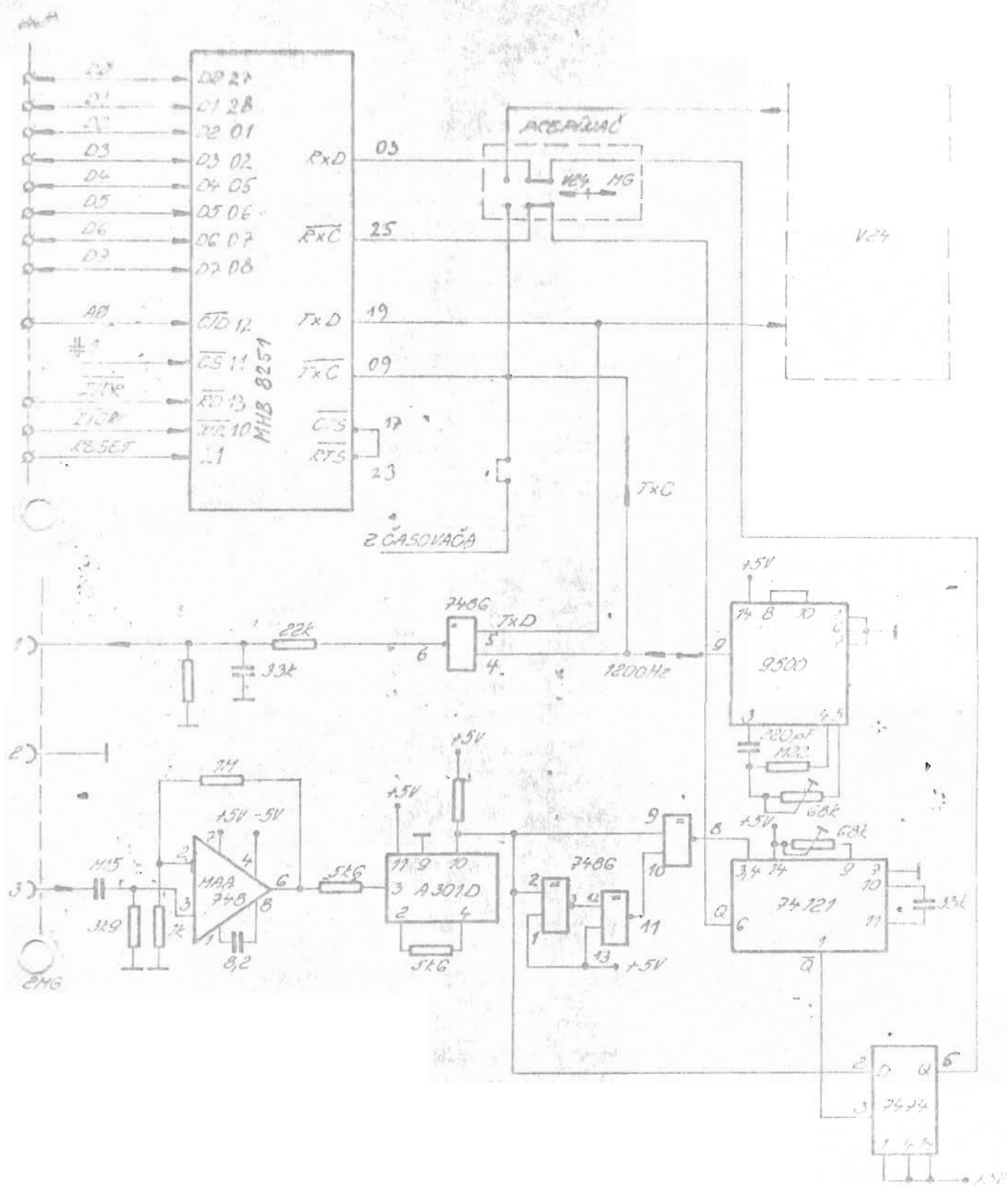


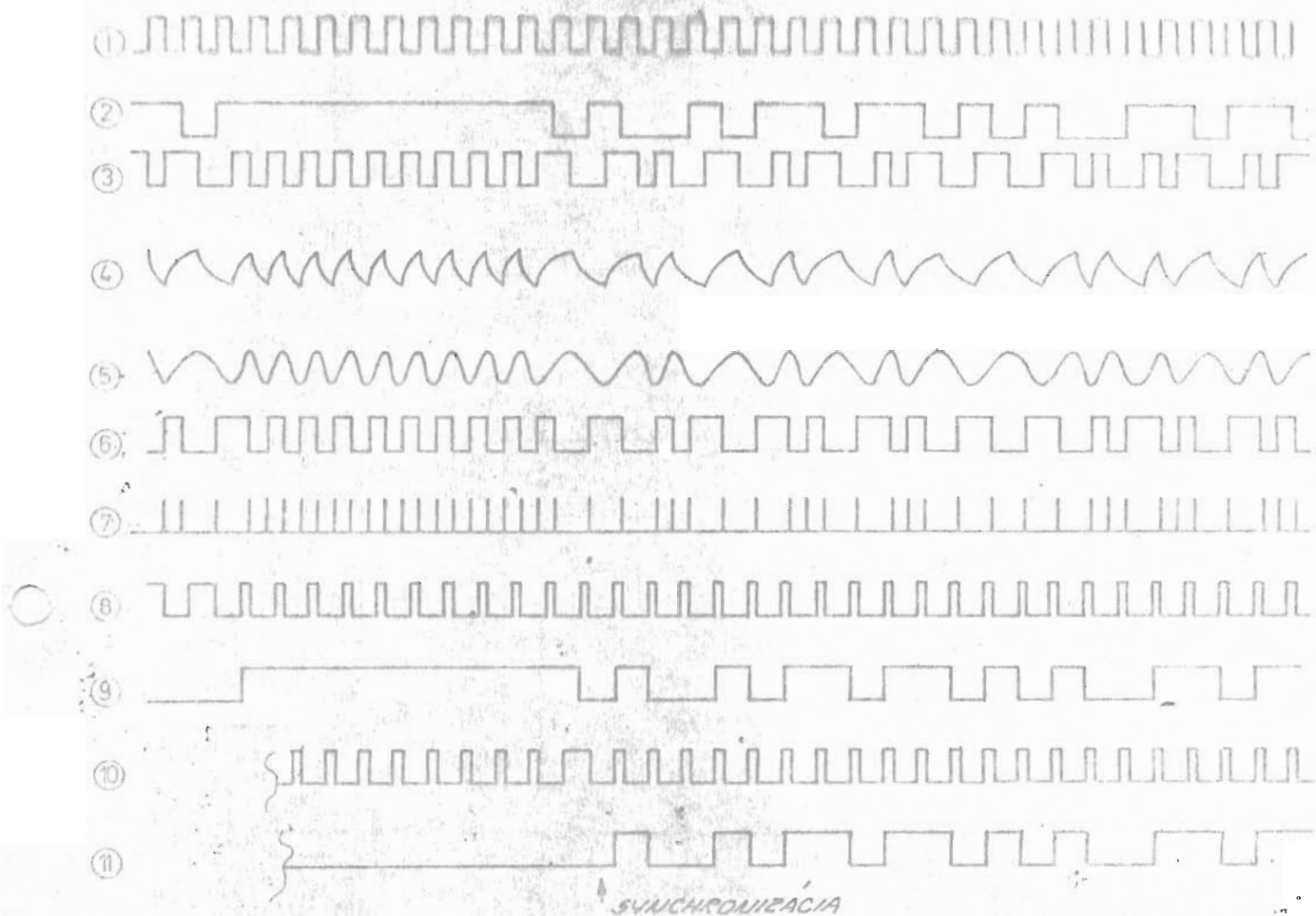
СХЕМА ЗАПІСКИ ІНТЕРФЕЙСУ ІМС-2 (HP-IB)



obr. č. 21 SCHEMA ZAPOJENIA INTERFEJSU V.24



obr. č. 22 SCHÉMA ZAPOJENIA INTERFEJSU PRE MAGNETOFÓN



① HODINY PRE USART (VYSIELAČ)

⑦ SPÚŠŤANIE HODÍN

② DÁTA Z USARTU

⑧ PLATNÉ HODINY

③ MODULOVANÉ DÁTA

⑨ PLATNÉ DÁTA PRE USART (PRIJÍMAČ)

④ SIGNÁL DO MGF

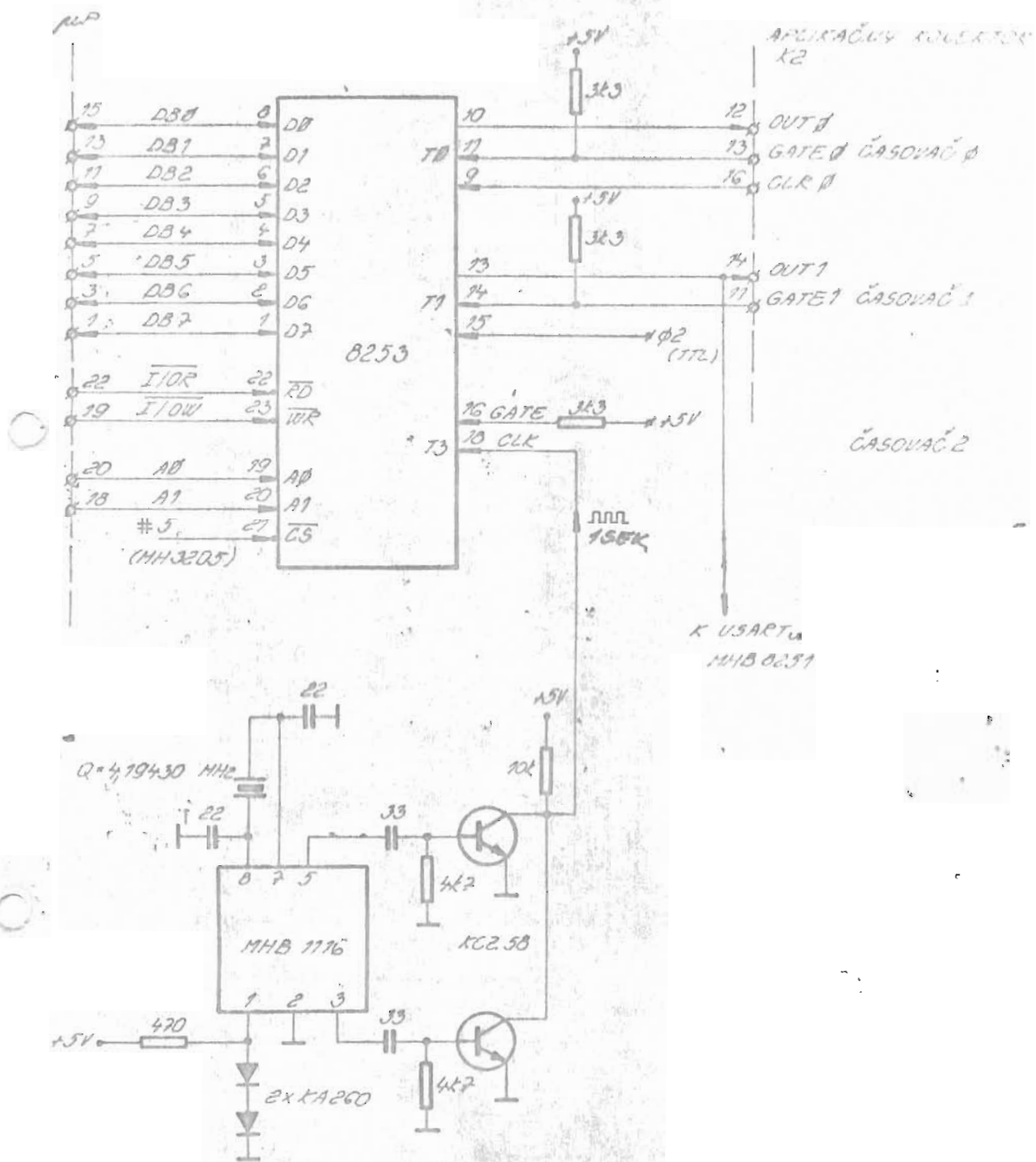
⑩ NEPLATNÉ HODINY A ICH SYNCHRONIZÁCIA

⑤ SIGNÁL Z MGF

⑪ NEPLATNÉ DÁTA A ICH SYNCHRONIZÁCIA

⑥ SIGNÁL Z OBMEDZOVAČA

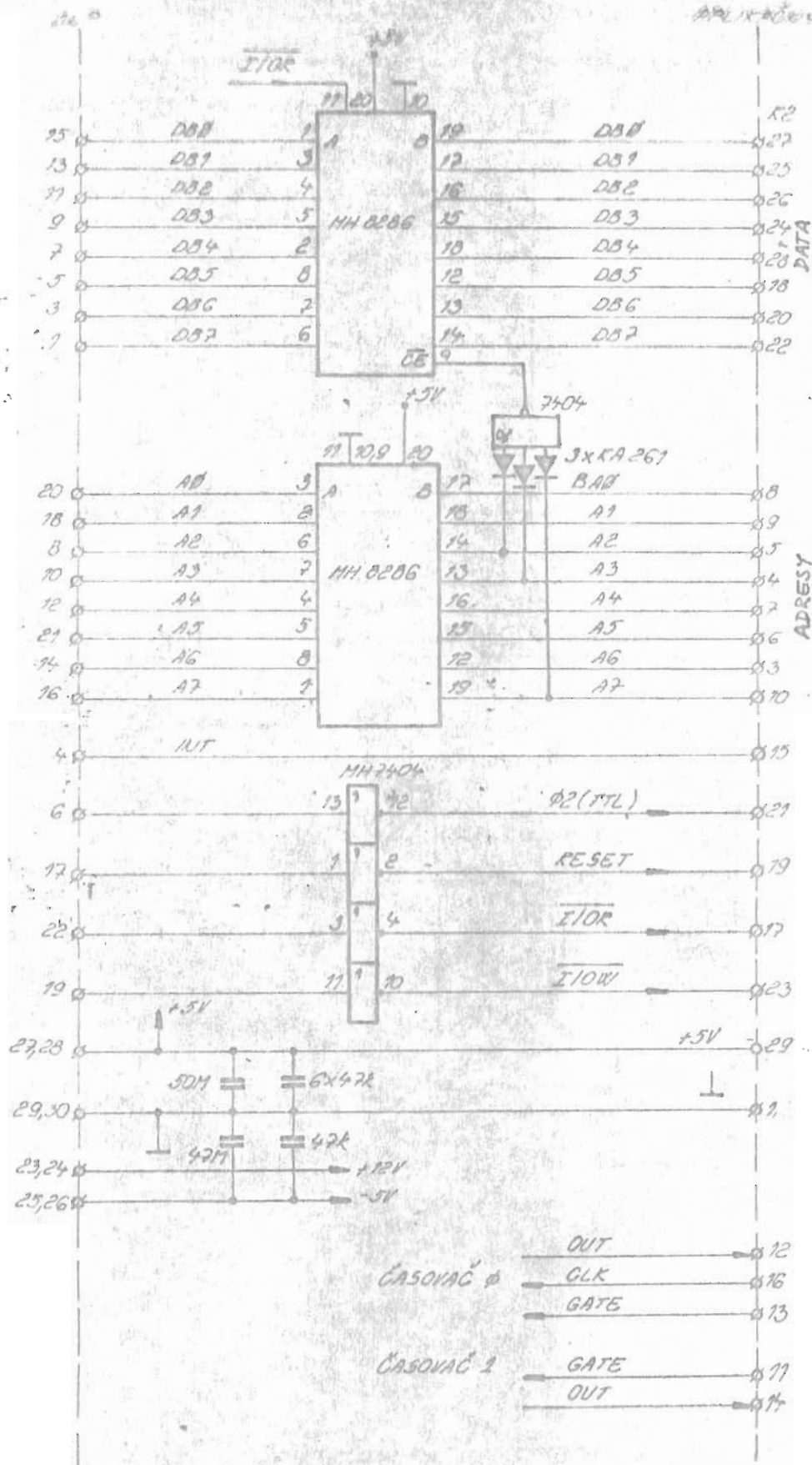
POZNÁMKA: PRE PRIJÍMAČ USART-U SÚ HODINY ⑧ ⑩ INVERTOVANÉ



obr. č. 24 SCHÉMA ZAPOJENIA INTERFEJSU ČASOVAČA

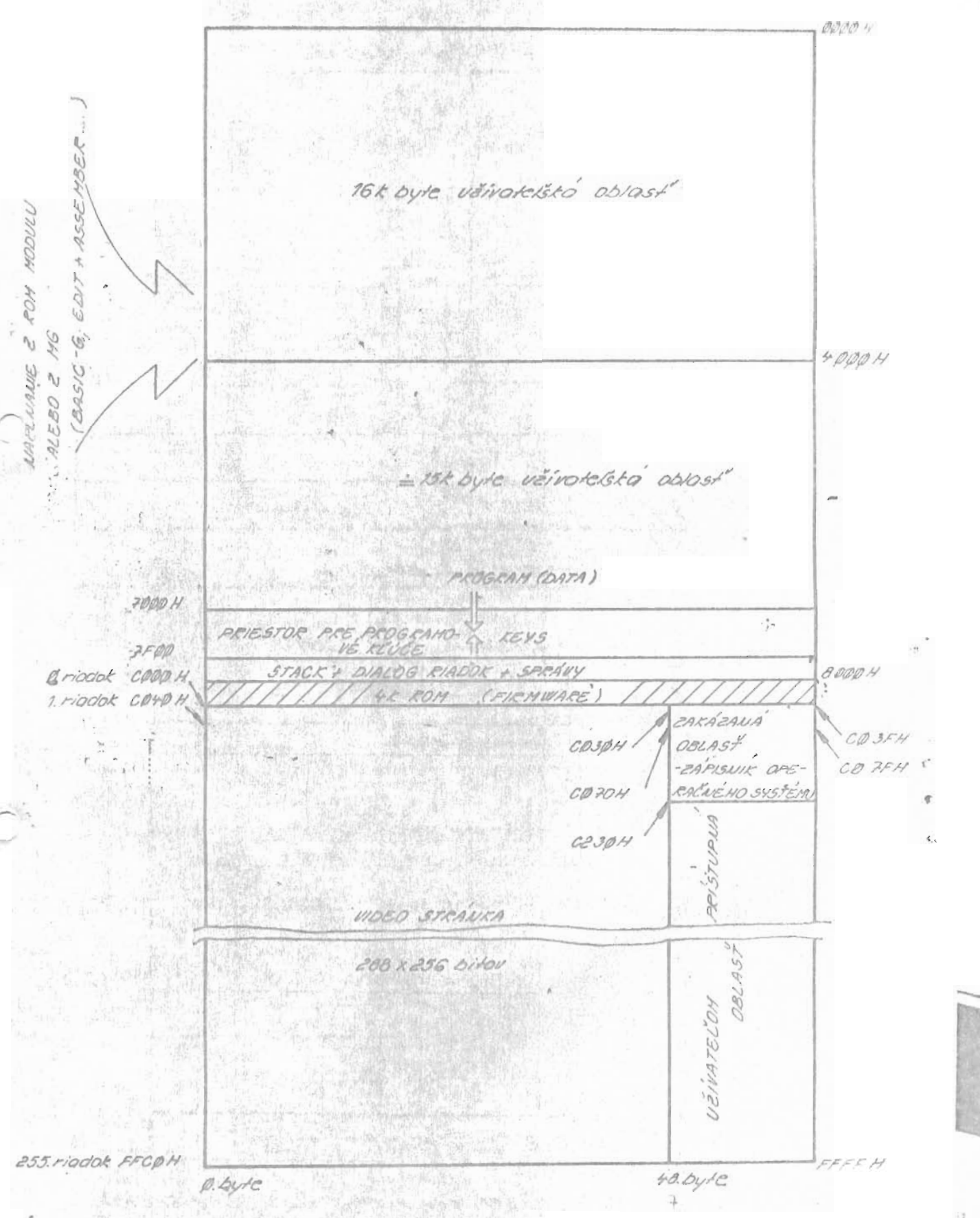
STANDARDNÍ KONEXE

APLIKACNÍ KONEXE

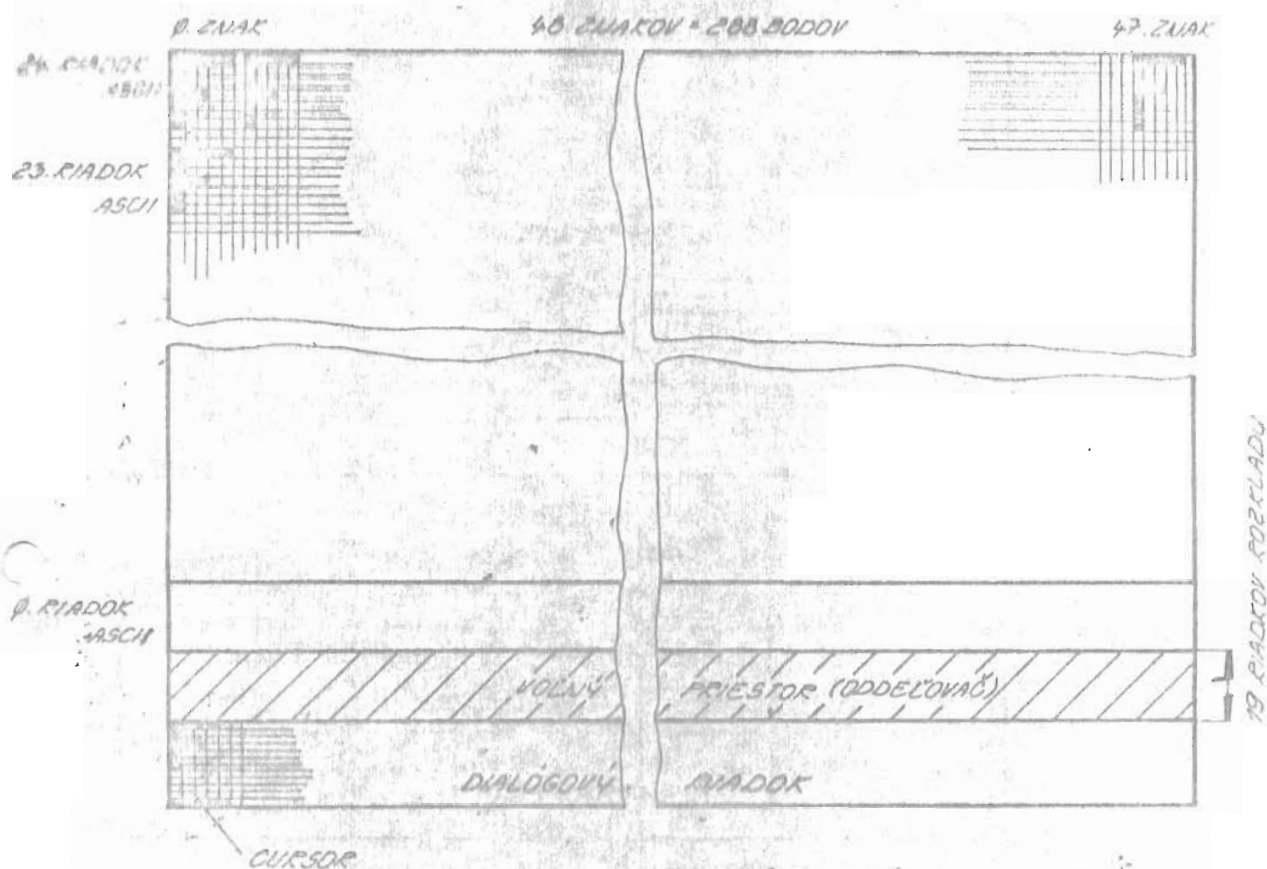


obr. č. 25 APLIKAČNÍ KONEXE

MAPUJEME 2 ROM MODULY
 ... ALEBO 2 MG
 (BASIC-G, EDIT + ASSEMBLER...)



obr. č. 27 ROZLOŽENIE PAMÄTE ROM



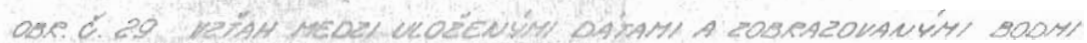
a) ALFANUMERICKÝ FORMÁT 48x25



b) GRAFICKÝ FORMÁT 288x243 (256x243)

* PLATÍ PRE FUNKCIE „BYTE“ ORIENTOVANÉ (POKE, BMOVE, BPLLOT...)

Obr. č. 28 FORMÁTY ORGANIZOVANIA VIDEO



Č. LÁZOV	UKAZATEĽ		ARGUMENT		VÝZNAM	POZNÁ. KA	
	HEX	DEC	HEX	DEC		PŘÍKLAD	
1. MODE POINT	C1FA	-15 878	AB 80 AF	168 176 175	REGACIA SET NUL	MODIFIKOVANIE BODOV V GRAFIKE	
2. CODE POINT (FARBA)	CO3A	-16 326	00 40 80 C0	0 64 128 192	100% 50% 100% 50% FBIK BIK	MODIFIKOVANIE FARBY ŠESTICE BODOV	
3. TASCII	CO3C CO3D	-16 324 -16 323	00 85	0 133	VIŠŠIA HODN. VIŠŠIA HODN.	UKAZATEĽ TABULIČKY PRE VÝSTUP ZNAKOV ASCII	
4. CURSOR	CO3E CO3F	-16 322 -16 321	00 FB	0 251	VIŠŠIA VIŠŠIA	UKAZATEĽ HODNOTY KURSORA, KTORÝ UKAZUJE MIESTO VÝPISU ASCII ZNAKOV	
5. BUFFER	* 7000	28 672	-	-	VOĽBA PAMÄTI	* VOĽBA PAMÄTOVÁ LOKÁCIA V ROZSAHU 3,5 BYTŮ	
6. P POINT	* 85E6	-31 258	-	-	ČÍTANIE BODU	* ADRESA KUTINY	
7. B READ	* 84F3	-31 301	-	-	ČÍTANIE ŠESTICE BODOV	* ADRESA KUTINY	
8. BELL	* 86	134	B1 B0 B2	1 0 2	TON 1 VYPNUTÉ TON 2	* ADRESA PORTU OVLÁDANIE AKUSTICKÉHO HLUČIA	
9. LED	* 86	134	B0 B4 B8	0 4 8	VYPNUTÉ LED 1 LED 2	* ADRESA PORTU OVLÁDANIE NUDKÁCIE	

OUT C. L. UKAZOVATEĽ A HODNOTY PRE MODIFIKACIU SYSTÉMU