

The ultimate Galaksija talk

Tomaž Šolc
tomaz.solc@tablix.org

29C3: 29th Chaos Communication Congress
30 December 2012





Cartography: Gasper Šubelj, 2011





Cartography: Gasper Šubelj 2011





Orao



Lola-8



Galeb









64-bitni računar "Galaksija"

Napravi i ti
Piše: Voja Antonić

Kada kupujete komercijalni model komputera, nalazite se u prilično prijateljskoj poziciji: dobijate računar koji se jednostavno povezuje sa televizorom, uključujete ga u mrežu i počinjete sa radom. Kada sami pravite računar, stvari su mnogo složenije: počinjete od neugledne gomile čipova, kondenzatora, otpornika i tastera i pokušavate da ih povežete u celinu koja će raditi. Ako budete pažljivo sledili uputstva koje ćemo dati, to ne bi trebalo da bude previše teško: nikakvo posebno predznanje, osim određene iskustva sa lemljenjem, nije potrebno da bi se, posle određenog broja časova rada, na ekranu vašeg televizora pojavilo ohrabrujuće READY.

Samogradnja, ipak, sa sobom nosi i određene rizike. Iako je računar „Galaksija“ izrađen već u osam primera koji besprekorno funkcionišu, pa nema dilema o ispravnosti samog projekta, postoji mogućnost da naiđete na neki neispravan čip, da nešto pogrešno postavite, napravite „mikroskopi“ kratak spoj na štampanom kolu ili stradate na nekoj drugoj „krivini“. Tada će biti neophodno nešto više znanja ili (u ekstremnom slučaju) stručna pomoć da bi se neispravnost pronašla i otklonila. Ipak, stara poslovica kaže da bez rizika nema ni uspeha. Računar koji ste napravili sopstvenim rukama i koji je proradio pružilo vam je zadovoljstvo od nekog komercijalnog modela, ne računajući činjenicu da ćete u toku gradnje videti iz čega se kompjuter sastoji i pronaći bar u neke tajne njegovog funkcionisanja.

Pre nego što donesete odluku o samogradnji računera, treba da razumete njegovu osnovnu koncepciju. Računar „Galaksija“ je možda posluži kao most između ljubitelja računara širom Jugoslavije.



Moćniji nego što izgleda. Računar „Galaksija“ predstavlja idealnu mašinu za učenje programiranja, ozbiljne primene u svakodnevnom životu i, svakako, zabavu. Pravi preko hiljadu naličica što znači da će uveličati razmena programa, informacija, kao i eventualno onovremeno ključova. Iz iskustva znamo da saradnja vlasnika različitih kompjutera (pa ma koliko ti kompjuteri bili moćni) ne može da bude naročito interesantna i plodna pa će računar „Galaksija“ možda poslužiti kao most između ljubitelja računara širom Jugoslavije.

Hardverska ...

Jednostavno štampano kolo

Računar „Galaksija“ se sastoji od štampanog kola, devetnaest ili dvadeset integriranih kola (zavisno da li se odlučite da stavite treći čip 6116 i tako proširite RAM memoriju „Galaksija“ na 6 kilobajta), određenog broja tranzistora, dioda, otpornika i kondenzatora (precizni popis opreme dajemo nešto donije), tastature, kučije i ispravljača. Štampano kolo je jednostavno, ali da ima manje mogućnosti za greške i da je, na kraju, smetanje elemenata povezano sa manjim rizikom. Jedna od posledica je da smo morali da upotrebimo priličan broj kratkospojnika („štampara“), što ne bi trebalo da izazove nikakve probleme osim izvesnog produženja pozora.

Softverski video

Jednolična štampa onemogućava postavljanje previde integriranih kola (čipova). Zeleni smo uz to, da računar „Galaksija“ bude što jeftiniji, pa smo morali da ne bi trebalo da se što više stvari reši programski

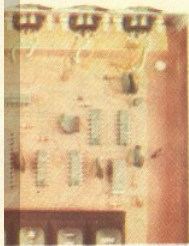
(softverski), a da broj čipova bude minimalan. Najveće pojednostavljenje hardvera je postignuto time što je video podržan softverski: mikroprocresor, naime, posle svake dve stotine sekunde prekida redovan posao i pomaže video-stepenu da ispiše slike na ekranu.

Statičke memorije

Da bi se shema računara maksimalno pojednostavila, upotrebljene su statičke umesto dinamičke popularnih dinamičkih memorija (dinamičke memorije se, okvirno govoreći, sastoje od kondenzatora koji računara mora da „dopunjavaju“ u kratkim vremenskim intervalima). Osim toga, za korišćenje statičkih memorija nije potrebno dovoditi tri stabilisana napona napajanja, što u mnogome pojednostavljuje štampano kolo i, što je još važnije, značajno smanjuje verovatnoću da pogreške pri izradi (vrio je verovatno da računara sa dinamičkim memorijama ne bi proradio čak ni kod umereno iskusnog konstruktora). Statičke memorije su, uz to, skoro kompatibilne sa EPROM-om 2716, što znači da se deo RAM-a lako može zameniti ROM-om. Statičke memorije, na žalost, imaju i neke loše strane: skuplje su i kapacitet im je manji, ali nam se čini da je njihovom upotrebom dobijeno daleko više nego što je izgubljeno. Ozirom da RAM memorija računara „Galaksija“ može da se povećava do 54 kilobajta, iskusnijim konstruktorima je ostavljena mogućnost da upotrebe dinamičke memorije za to proširenje i tako uštede novac.

Standardni čipovi

Jedan od važnih konceptualnih zahteva bio je da se koriste standardni čipovi koji



prepostiti vama. Možete upotrebiti onakve čipove kao što su standardni DIP-čipovi (DIP-čipovi su čipovi sa 14, 16, 18, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 68, 72, 76, 80, 84, 88, 92, 96, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 144, 148, 152, 156, 160, 164, 168, 172, 176, 180, 184, 188, 192, 196, 200, 204, 208, 212, 216, 220, 224, 228, 232, 236, 240, 244, 248, 252, 256, 260, 264, 268, 272, 276, 280, 284, 288, 292, 296, 300, 304, 308, 312, 316, 320, 324, 328, 332, 336, 340, 344, 348, 352, 356, 360, 364, 368, 372, 376, 380, 384, 388, 392, 396, 400, 404, 408, 412, 416, 420, 424, 428, 432, 436, 440, 444, 448, 452, 456, 460, 464, 468, 472, 476, 480, 484, 488, 492, 496, 500, 504, 508, 512, 516, 520, 524, 528, 532, 536, 540, 544, 548, 552, 556, 560, 564, 568, 572, 576, 580, 584, 588, 592, 596, 600, 604, 608, 612, 616, 620, 624, 628, 632, 636, 640, 644, 648, 652, 656, 660, 664, 668, 672, 676, 680, 684, 688, 692, 696, 700, 704, 708, 712, 716, 720, 724, 728, 732, 736, 740, 744, 748, 752, 756, 760, 764, 768, 772, 776, 780, 784, 788, 792, 796, 800, 804, 808, 812, 816, 820, 824, 828, 832, 836, 840, 844, 848, 852, 856, 860, 864, 868, 872, 876, 880, 884, 888, 892, 896, 900, 904, 908, 912, 916, 920, 924, 928, 932, 936, 940, 944, 948, 952, 956, 960, 964, 968, 972, 976, 980, 984, 988, 992, 996, 1000, 1004, 1008, 1012, 1016, 1020, 1024, 1028, 1032, 1036, 1040, 1044, 1048, 1052, 1056, 1060, 1064, 1068, 1072, 1076, 1080, 1084, 1088, 1092, 1096, 1100, 1104, 1108, 1112, 1116, 1120, 1124, 1128, 1132, 1136, 1140, 1144, 1148, 1152, 1156, 1160, 1164, 1168, 1172, 1176, 1180, 1184, 1188, 1192, 1196, 1200, 1204, 1208, 1212, 1216, 1220, 1224, 1228, 1232, 1236, 1240, 1244, 1248, 1252, 1256, 1260, 1264, 1268, 1272, 1276, 1280, 1284, 1288, 1292, 1296, 1300, 1304, 1308, 1312, 1316, 1320, 1324, 1328, 1332, 1336, 1340, 1344, 1348, 1352, 1356, 1360, 1364, 1368, 1372, 1376, 1380, 1384, 1388, 1392, 1396, 1400, 1404, 1408, 1412, 1416, 1420, 1424, 1428, 1432, 1436, 1440, 1444, 1448, 1452, 1456, 1460, 1464, 1468, 1472, 1476, 1480, 1484, 1488, 1492, 1496, 1500, 1504, 1508, 1512, 1516, 1520, 1524, 1528, 1532, 1536, 1540, 1544, 1548, 1552, 1556, 1560, 1564, 1568, 1572, 1576, 1580, 1584, 1588, 1592, 1596, 1600, 1604, 1608, 1612, 1616, 1620, 1624, 1628, 1632, 1636, 1640, 1644, 1648, 1652, 1656, 1660, 1664, 1668, 1672, 1676, 1680, 1684, 1688, 1692, 1696, 1700, 1704, 1708, 1712, 1716, 1720, 1724, 1728, 1732, 1736, 1740, 1744, 1748, 1752, 1756, 1760, 1764, 1768, 1772, 1776, 1780, 1784, 1788, 1792, 1796, 1800, 1804, 1808, 1812, 1816, 1820, 1824, 1828, 1832, 1836, 1840, 1844, 1848, 1852, 1856, 1860, 1864, 1868, 1872, 1876, 1880, 1884, 1888, 1892, 1896, 1900, 1904, 1908, 1912, 1916, 1920, 1924, 1928, 1932, 1936, 1940, 1944, 1948, 1952, 1956, 1960, 1964, 1968, 1972, 1976, 1980, 1984, 1988, 1992, 1996, 2000, 2004, 2008, 2012, 2016, 2020, 2024, 2028, 2032, 2036, 2040, 2044, 2048, 2052, 2056, 2060, 2064, 2068, 2072, 2076, 2080, 2084, 2088, 2092, 2096, 2100, 2104, 2108, 2112, 2116, 2120, 2124, 2128, 2132, 2136, 2140, 2144, 2148, 2152, 2156, 2160, 2164, 2168, 2172, 2176, 2180, 2184, 2188, 2192, 2196, 2200, 2204, 2208, 2212, 2216, 2220, 2224, 2228, 2232, 2236, 2240, 2244, 2248, 2252, 2256, 2260, 2264, 2268, 2272, 2276, 2280, 2284, 2288, 2292, 2296, 2300, 2304, 2308, 2312, 2316, 2320, 2324, 2328, 2332, 2336, 2340, 2344, 2348, 2352, 2356, 2360, 2364, 2368, 2372, 2376, 2380, 2384, 2388, 2392, 2396, 2400, 2404, 2408, 2412, 2416, 2420, 2424, 2428, 2432, 2436, 2440, 2444, 2448, 2452, 2456, 2460, 2464, 2468, 2472, 2476, 2480, 2484, 2488, 2492, 2496, 2500, 2504, 2508, 2512, 2516, 2520, 2524, 2528, 2532, 2536, 2540, 2544, 2548, 2552, 2556, 2560, 2564, 2568, 2572, 2576, 2580, 2584, 2588, 2592, 2596, 2600, 2604, 2608, 2612, 2616, 2620, 2624, 2628, 2632, 2636, 2640, 2644, 2648, 2652, 2656, 2660, 2664, 2668, 2672, 2676, 2680, 2684, 2688, 2692, 2696, 2700, 2704, 2708, 2712, 2716, 2720, 2724, 2728, 2732, 2736, 2740, 2744, 2748, 2752, 2756, 2760, 2764, 2768, 2772, 2776, 2780, 2784, 2788, 2792, 2796, 2800, 2804, 2808, 2812, 2816, 2820, 2824, 2828, 2832, 2836, 2840, 2844, 2848, 2852, 2856, 2860, 2864, 2868, 2872, 2876, 2880, 2884, 2888, 2892, 2896, 2900, 2904, 2908, 2912, 2916, 2920, 2924, 2928, 2932, 2936, 2940, 2944, 2948, 2952, 2956, 2960, 2964, 2968, 2972, 2976, 2980, 2984, 2988, 2992, 2996, 3000, 3004, 3008, 3012, 3016, 3020, 3024, 3028, 3032, 3036, 3040, 3044, 3048, 3052, 3056, 3060, 3064, 3068, 3072, 3076, 3080, 3084, 3088, 3092, 3096, 3100, 3104, 3108, 3112, 3116, 3120, 3124, 3128, 3132, 3136, 3140, 3144, 3148, 3152, 3156, 3160, 3164, 3168, 3172, 3176, 3180, 3184, 3188, 3192, 3196, 3200, 3204, 3208, 3212, 3216, 3220, 3224, 3228, 3232, 3236, 3240, 3244, 3248, 3252, 3256, 3260, 3264, 3268, 3272, 3276, 3280, 3284, 3288, 3292, 3296, 3300, 3304, 3308, 3312, 3316, 3320, 3324, 3328, 3332, 3336, 3340, 3344, 3348, 3352, 3356, 3360, 3364, 3368, 3372, 3376, 3380, 3384, 3388, 3392, 3396, 3400, 3404, 3408, 3412, 3416, 3420, 3424, 3428, 3432, 3436, 3440, 3444, 3448, 3452, 3456, 3460, 3464, 3468, 3472, 3476, 3480, 3484, 3488, 3492, 3496, 3500, 3504, 3508, 3512, 3516, 3520, 3524, 3528, 3532, 3536, 3540, 3544, 3548, 3552, 3556, 3560, 3564, 3568, 3572, 3576, 3580, 3584, 3588, 3592, 3596, 3600, 3604, 3608, 3612, 3616, 3620, 3624, 3628, 3632, 3636, 3640, 3644, 3648, 3652, 3656, 3660, 3664, 3668, 3672, 3676, 3680, 3684, 3688, 3692, 3696, 3700, 3704, 3708, 3712, 3716, 3720, 3724, 3728, 3732, 3736, 3740, 3744, 3748, 3752, 3756, 3760, 3764, 3768, 3772, 3776, 3780, 3784, 3788, 3792, 3796, 3800, 3804, 3808, 3812, 3816, 3820, 3824, 3828, 3832, 3836, 3840, 3844, 3848, 3852, 3856, 3860, 3864, 3868, 3872, 3876, 3880, 3884, 3888, 3892, 3896, 3900, 3904, 3908, 3912, 3916, 3920, 3924, 3928, 3932, 3936, 3940, 3944, 3948, 3952, 3956, 3960, 3964, 3968, 3972, 3976, 3980, 3984, 3988, 3992, 3996, 4000, 4004, 4008, 4012, 4016, 4020, 4024, 4028, 4032, 4036, 4040, 4044, 4048, 4052, 4056, 4060, 4064, 4068, 4072, 4076, 4080, 4084, 4088, 4092, 4096, 4100, 4104, 4108, 4112, 4116, 4120, 4124, 4128, 4132, 4136, 4140, 4144, 4148, 4152, 4156, 4160, 4164, 4168, 4172, 4176, 4180, 4184, 4188, 4192, 4196, 4200, 4204, 4208, 4212, 4216, 4220, 4224, 4228, 4232, 4236, 4240, 4244, 4248, 4252, 4256, 4260, 4264, 4268, 4272, 4276, 4280, 4284, 4288, 4292, 4296, 4300, 4304, 4308, 4312, 4316, 4320, 4324, 4328, 4332, 4336, 4340, 4344, 4348, 4352, 4356, 4360, 4364, 4368, 4372, 4376, 4380, 4384, 4388, 4392, 4396, 4400, 4404, 4408, 4412, 4416, 4420, 4424, 4428, 4432, 4436, 4440, 4444, 4448, 4452, 4456, 4460, 4464, 4468, 4472, 4476, 4480, 4484, 4488, 4492, 4496, 4500, 4504, 4508, 4512, 4516, 4520, 4524, 4528, 4532, 4536, 4540, 4544, 4548, 4552, 4556, 4560, 4564, 4568, 4572, 4576, 4580, 4584, 4588, 4592, 4596, 4600, 4604, 4608, 4612, 4616, 4620, 4624, 4628, 4632, 4636, 4640, 4644, 4648, 4652, 4656, 4660, 4664, 4668, 4672, 4676, 4680, 4684, 4688, 4692, 4696, 4700, 4704, 4708, 4712, 4716, 4720, 4724, 4728, 4732, 4736, 4740, 4744, 4748, 4752, 4756, 4760, 4764, 4768, 4772, 4776, 4780, 4784, 4788, 4792, 4796, 4800, 4804, 4808, 4812, 4816, 4820, 4824, 4828, 4832, 4836, 4840, 4844, 4848, 4852, 4856, 4860, 4864, 4868, 4872, 4876, 4880, 4884, 4888, 4892, 4896, 4900, 4904, 4908, 4912, 4916, 4920, 4924, 4928, 4932, 4936, 4940, 4944, 4948, 4952, 4956, 4960, 4964, 4968, 4972, 4976, 4980, 4984, 4988, 4992, 4996, 5000, 5004, 5008, 5012, 5016, 5020, 5024, 5028, 5032, 5036, 5040, 5044, 5048, 5052, 5056, 5060, 5064, 5068, 5072, 5076, 5080, 5084, 5088, 5092, 5096, 5100, 5104, 5108, 5112, 5116, 5120, 5124, 5128, 5132, 5136, 5140, 5144, 5148, 5152, 5156, 5160, 5164, 5168, 5172, 5176, 5180, 5184, 5188, 5192, 5196, 5200, 5204, 5208, 5212, 5216, 5220, 5224, 5228, 5232, 5236, 5240, 5244, 5248, 5252, 5256, 5260, 5264, 5268, 5272, 5276, 5280, 5284, 5288, 5292, 5296, 5300, 5304, 5308, 5312, 5316, 5320, 5324, 5328, 5332, 5336, 5340, 5344, 5348, 5352, 5356, 5360, 5364, 5368, 5372, 5376, 5380, 5384, 5388, 5392, 5396, 5400, 5404, 5408, 5412, 5416, 5420, 5424, 5428, 5432, 5436, 5440, 5444, 5448, 5452, 5456, 5460, 5464, 5468, 5472, 5476, 5480, 5484, 5488, 5492, 5496, 5500, 5504, 5508, 5512, 5516, 5520, 5524, 5528, 5532, 5536, 5540, 5544, 5548, 5552, 5556, 5560, 5564, 5568, 5572, 5576, 5580, 5584, 5588, 5592, 5596, 5600, 5604, 5608, 5612, 5616, 5620, 5624, 5628, 5632, 5636, 5640, 5644, 5648, 5652, 5656, 5660, 5664, 5668, 5672, 5676, 5680, 5684, 5688, 5692, 5696, 5700, 5704, 5708, 5712, 5716, 5720, 5724, 5728, 5732, 5736, 5740, 5744, 5748, 5752, 5756, 5760, 5764, 5768, 5772, 5776, 5780, 5784, 5788, 5792, 5796, 5800, 5804, 5808, 5812, 5816, 5820, 5824, 5828, 5832, 5836, 5840, 5844, 5848, 5852, 5856, 5860, 5864, 5868, 5872, 5876, 5880, 5884, 5888, 5892, 5896, 5900, 5904, 5908, 5912, 5916, 5920, 5924, 5928, 5932, 5936, 5940, 5944, 5948, 5952, 5956, 5960, 5964, 5968, 5972, 5976, 5980, 5984, 5988, 5992, 5996, 6000, 6004, 6008, 6012, 6016, 6020, 6024, 6028, 6032, 6036, 6040, 6044, 6048, 6052, 6056, 6060, 6064, 6068, 6072, 6076, 6080, 6084, 6088, 6092, 6096, 6100, 6104, 6108, 6112, 6116, 6120, 6124, 6128, 6132, 6136, 6140, 6144, 6148, 6152, 6156, 6160, 6164, 6168, 6172, 6176, 6180, 6184, 6188, 6192, 6196, 6200, 6204, 6208, 6212, 6216, 6220, 6224, 6228, 6232, 6236, 6240, 6244, 6248, 6252, 6256, 6260, 6264, 6268, 6272, 6276, 6280, 6284, 6288, 6292, 6296, 6300, 6304, 6308, 6312, 6316, 6320, 6324, 6328, 6332, 6336, 6340, 6344, 6348, 6352, 6356, 6360, 6364, 6368, 6372, 6376, 6380, 6384, 6388, 6392, 6396, 6400, 6404, 6408, 6412, 6416, 6420, 6424, 6428, 6432, 6436, 6440, 6444, 6448, 6452, 6456, 6460, 6464, 6468, 6472, 6476, 6480, 6484, 6488, 6492, 6496, 6500, 6504, 6508, 6512, 6516, 6520, 6524, 6528, 6532, 6536, 6540, 6544, 6548, 6552, 6556, 6560, 6564, 6568, 6572, 6576, 6580, 6584, 6588, 6592, 6596, 6600, 6604, 6608, 6612, 6616, 6620, 6624, 6628, 6632, 6636, 6640, 6644, 6648, 6652, 6656, 6660, 6664, 6668, 6672, 6676, 6680, 6684, 6688, 6692, 6696, 6700, 6704, 6708, 6712, 6716, 6720, 6724, 6728, 6732, 6736, 6740, 6744, 6748, 6752, 6756, 6760, 6764, 6768, 6772, 6776, 6780, 6784, 6788, 6792, 6796, 6800, 6804, 6808, 6812, 6816, 6820, 6824, 6828, 6832, 6836, 6840, 6844, 6848, 6852, 6856, 6860, 6864, 6868, 6872, 6876, 6880, 6884, 6888, 6892, 6896, 6900, 6904, 6908, 6912, 6916, 6920, 6924, 6928, 6932, 6936, 6940, 6944, 6948, 6952, 6956, 6960, 6964, 6968, 697







Proje
Pa
6

Proce
1
R
R
Teks
Grafika
Boje
Zvuk

Maker: 6
Origin: Sa
Year: 198
Language: BRA
CPU: Zilac
Speed: 3.072
RAM: 2-6 KB
ROM: 4 KB
Text: 32x16
Graphics: 64x48
Colours: mono
Sound: ✓





Zilog Z80A @ 3.072 MHz



Zilog Z80A @ 3.072 MHz

**ROM A: 4 kB
ROM B: 4 kB**



Zilog Z80A @ 3.072 MHz

**ROM A: 4 kB
ROM B: 4 kB**

RAM: 6 kB

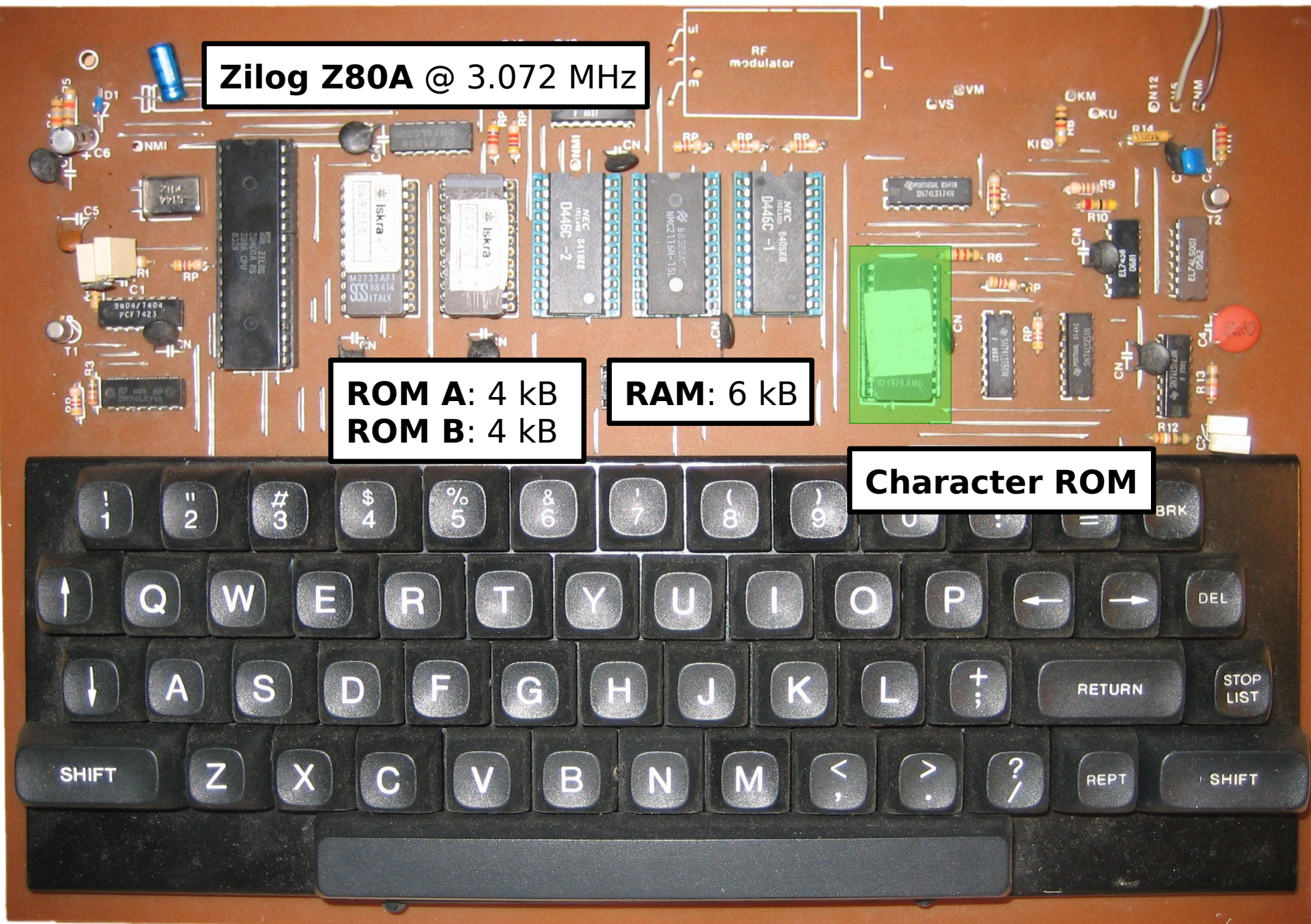


Zilog Z80A @ 3.072 MHz

**ROM A: 4 kB
ROM B: 4 kB**

RAM: 6 kB

Character ROM



Zilog Z80A @ 3.072 MHz

**6-bit register
(latch)**

**ROM A: 4 kB
ROM B: 4 kB**

RAM: 6 kB

Character ROM



Zilog Z80A @ 3.072 MHz

**6-bit register
(latch)**

**13 74LS-series
digital ICs**

**ROM A: 4 kB
ROM B: 4 kB**

RAM: 6 kB

Character ROM



5V DC power



5V



IN/OUT



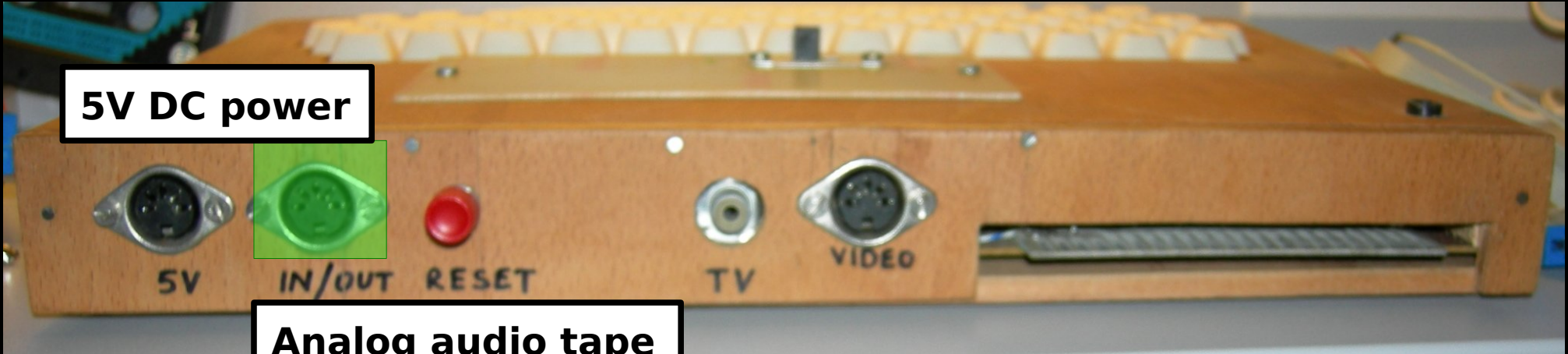
RESET



TV



VIDEO



5V DC power

The image shows the front panel of a wooden electronic device. From left to right, it features a 5V DC power port, a green IN/OUT port, a red RESET button, a TV port, and a VIDEO port. A keyboard is visible in the background.

IN/OUT

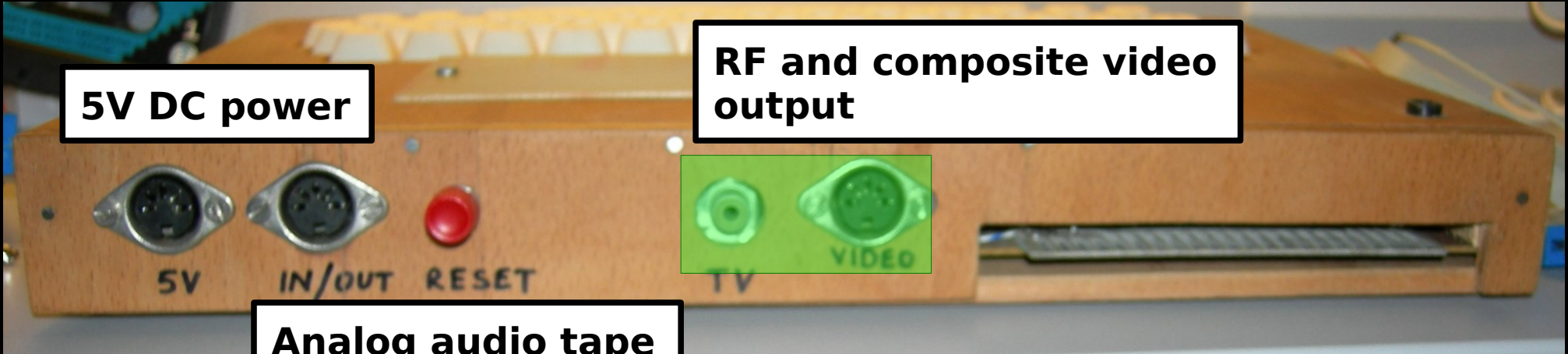
RESET

TV

VIDEO

5V

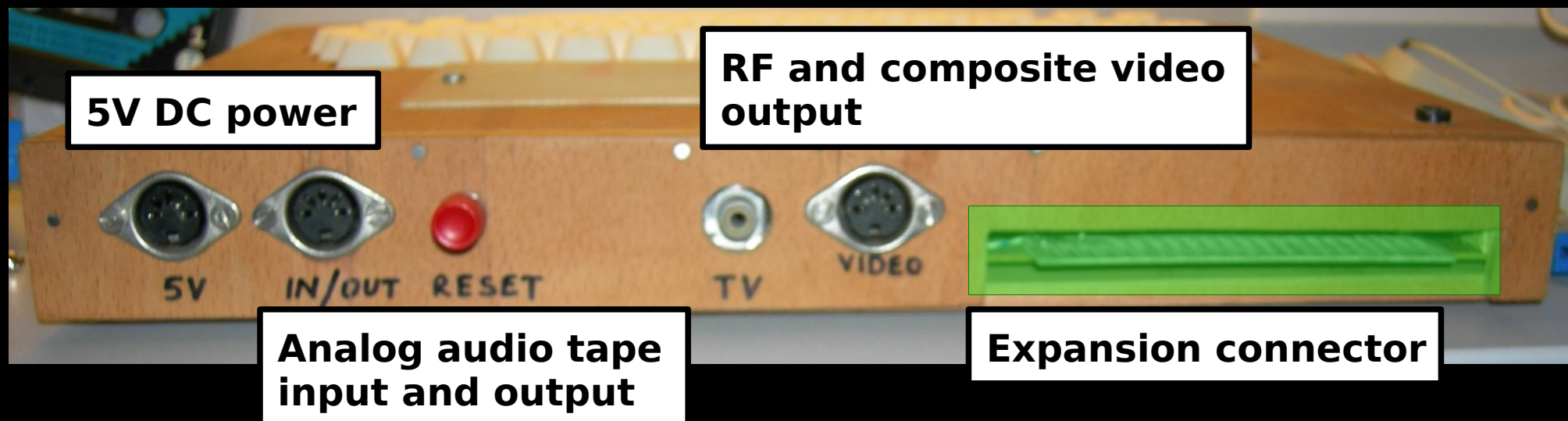
Analogue audio tape
input and output



5V DC power

RF and composite video output

Analog audio tape input and output



⇔ READY

> _

⇐ READY

>10 PRINT "HELLO, WORLD!"

>20 GOTO 10

>_

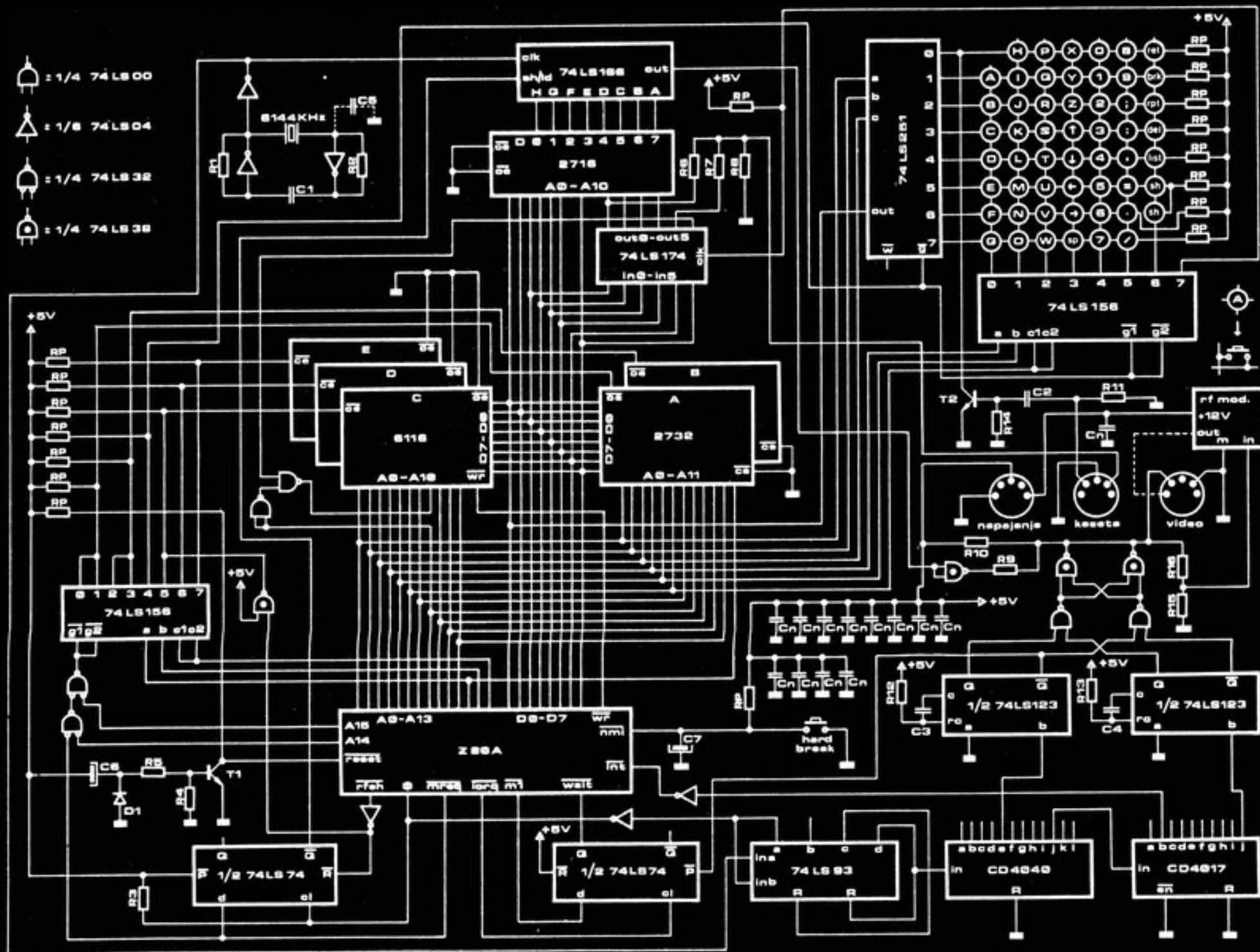
⇄ READY
>OLD_

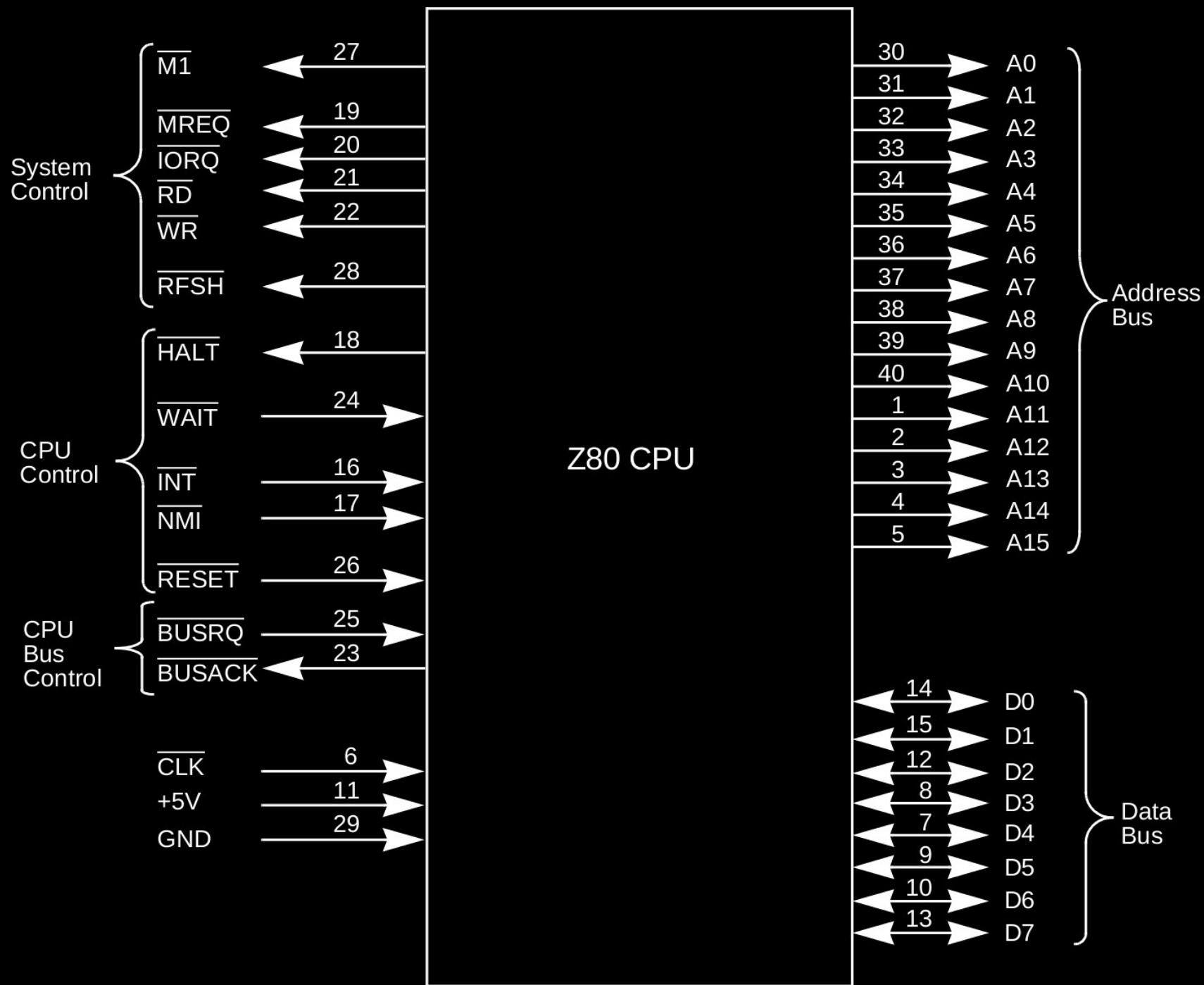
⇌ READY

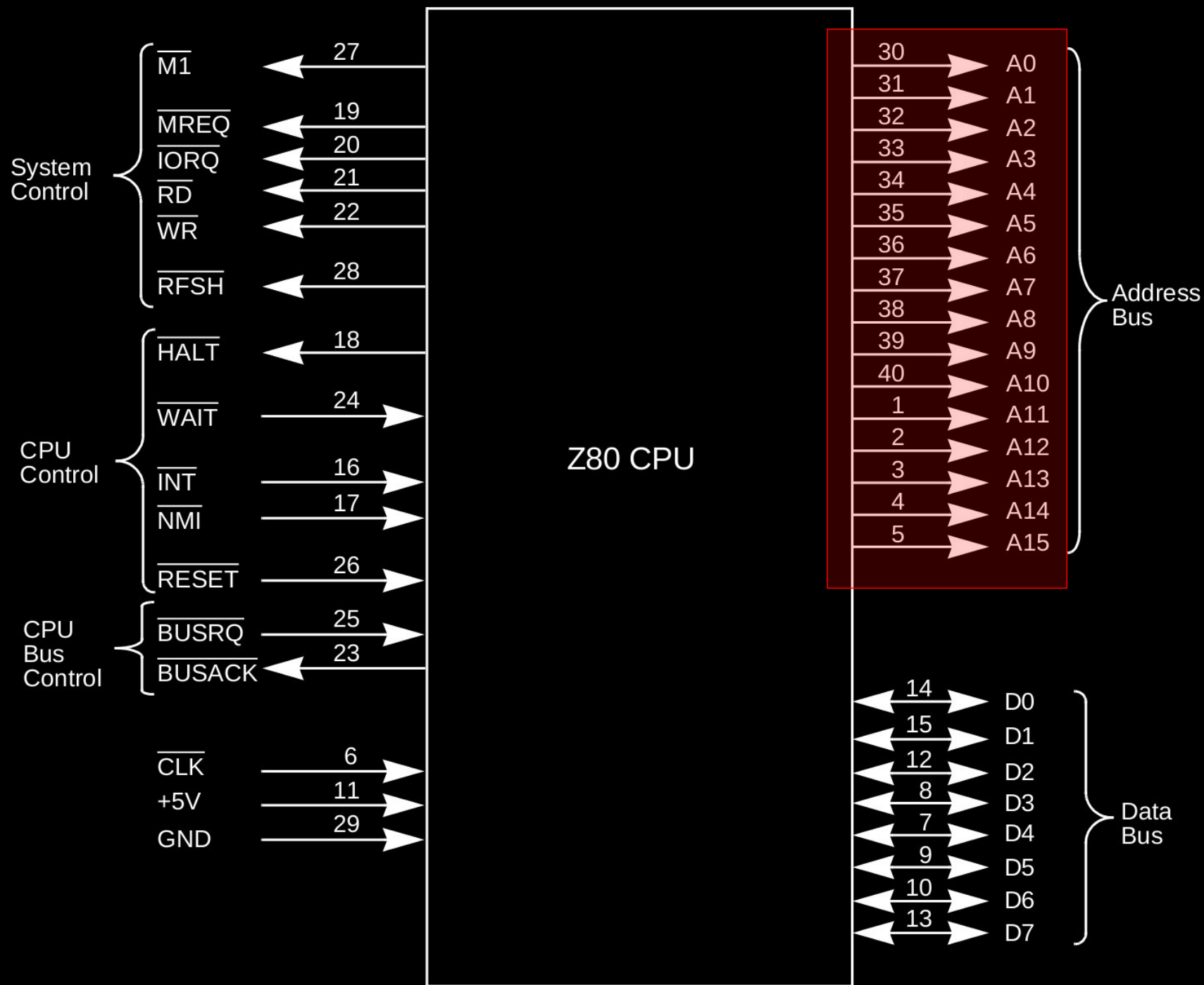
>OLD

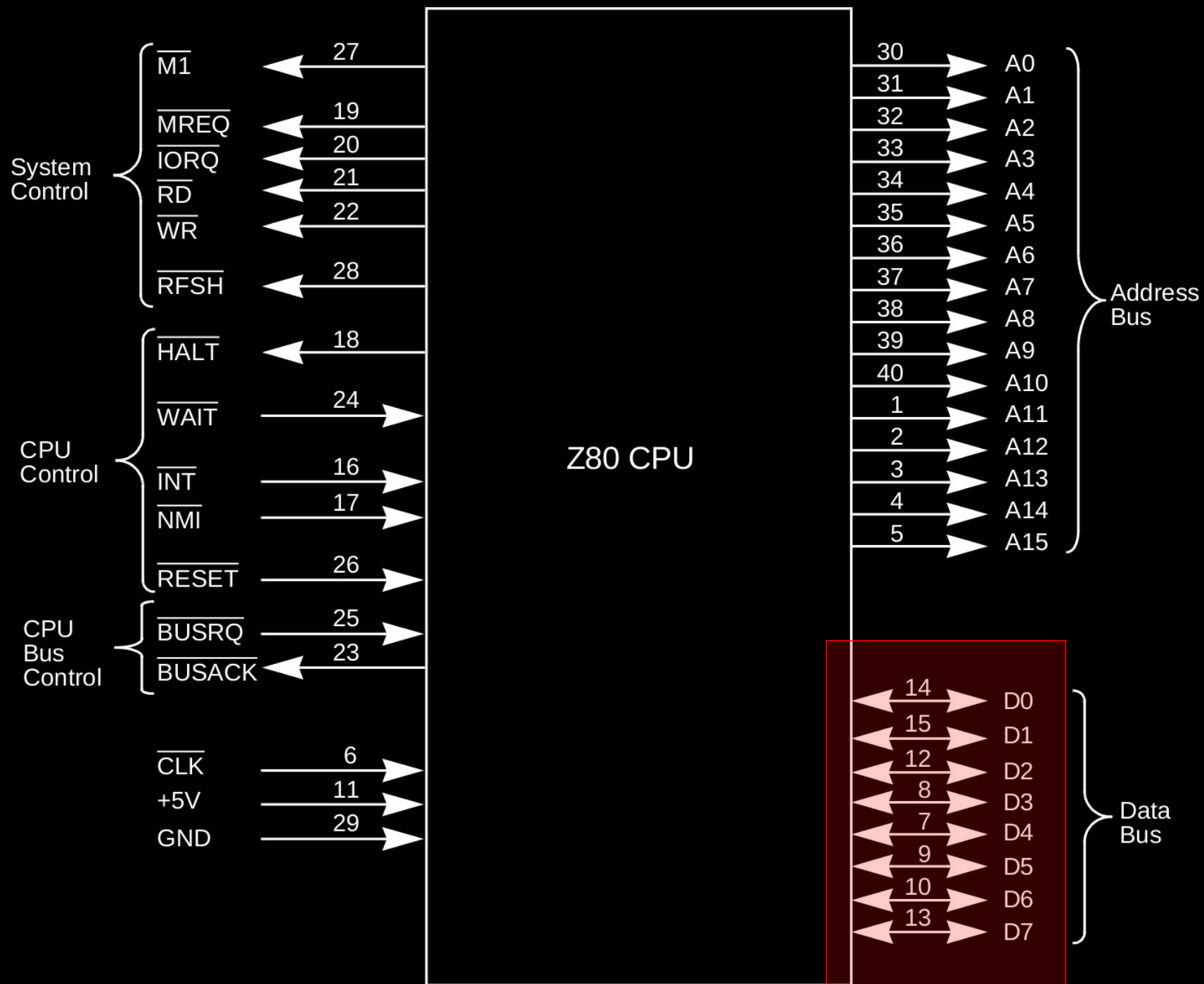
⇌ READY

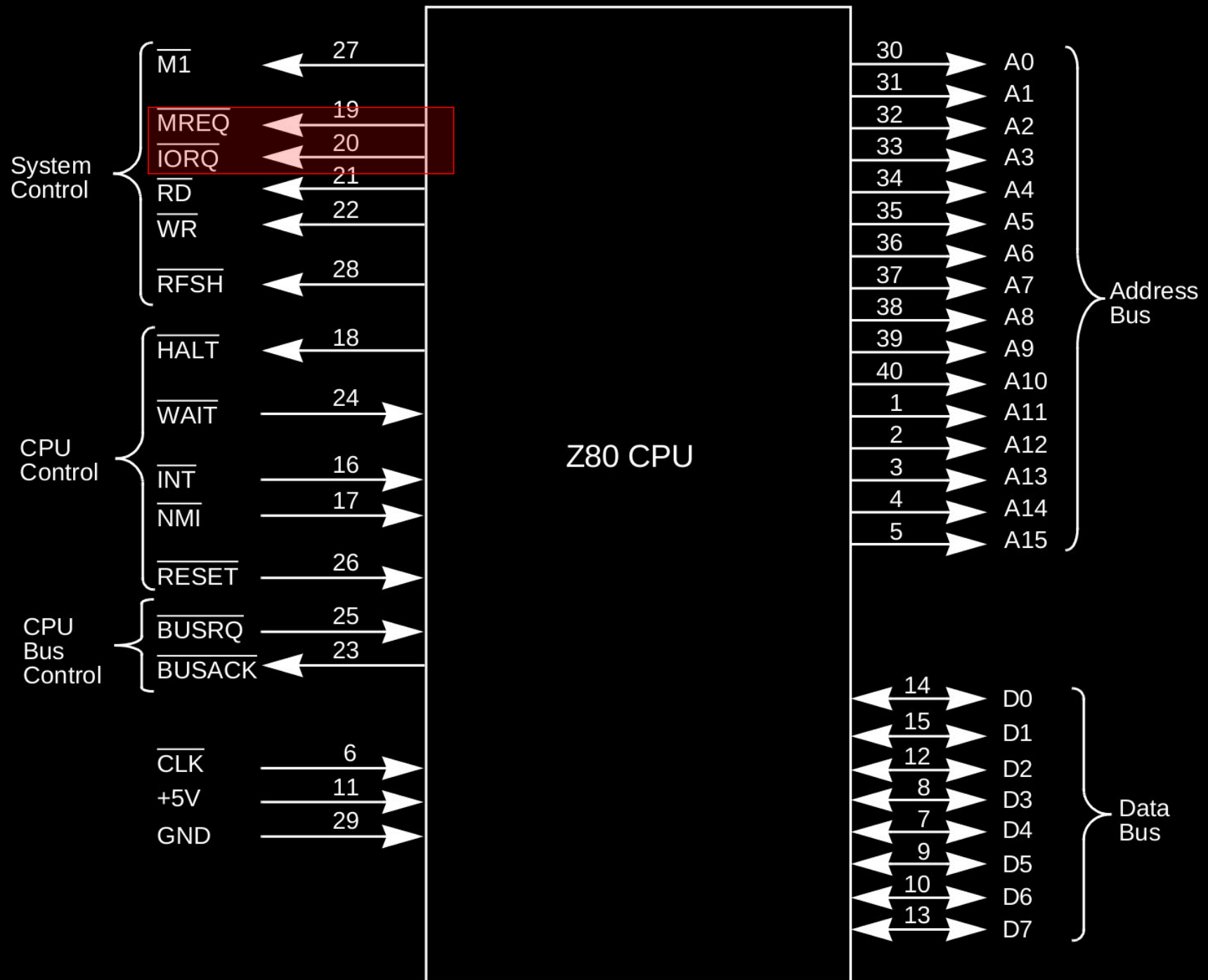
>RUN_

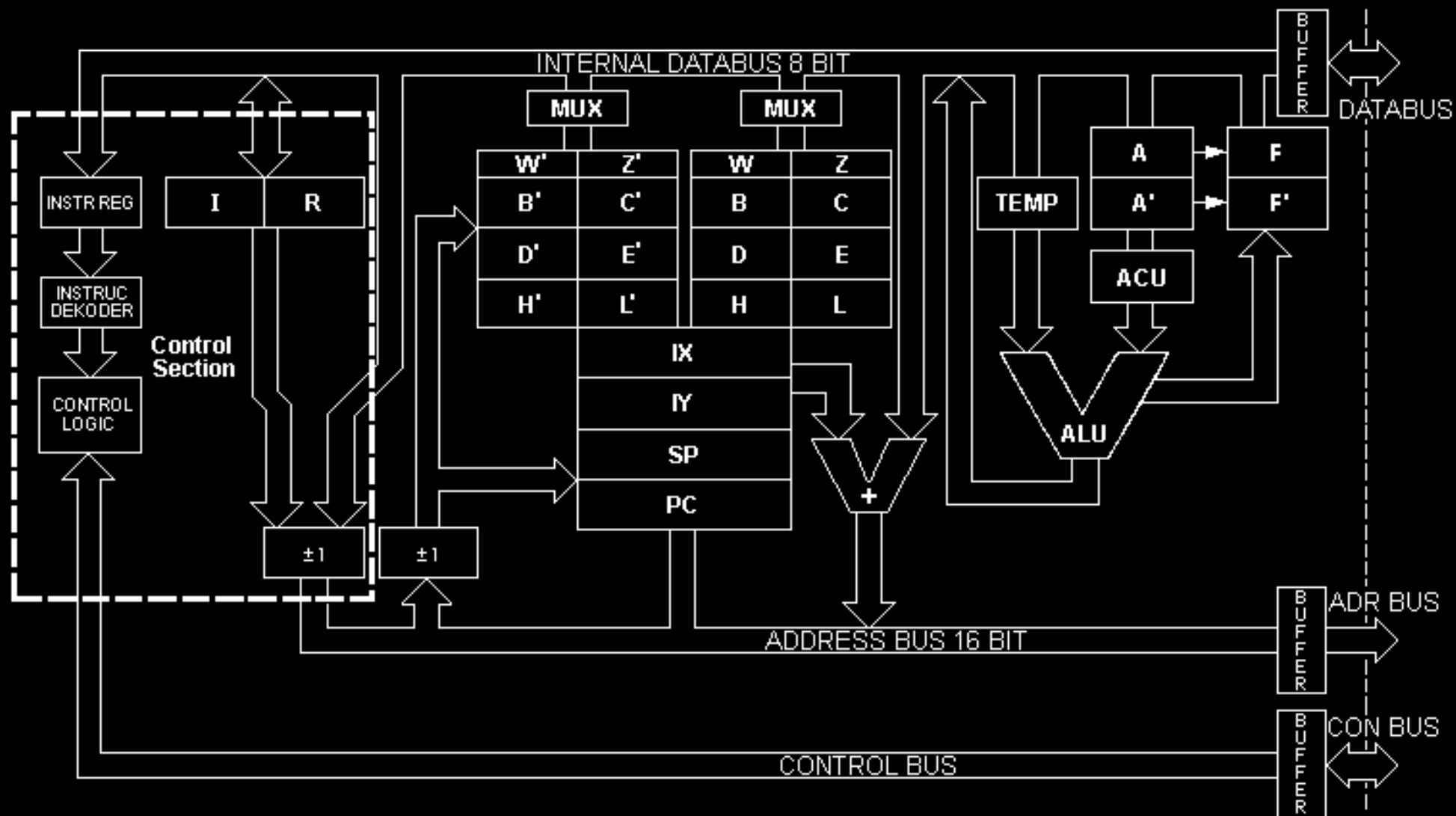


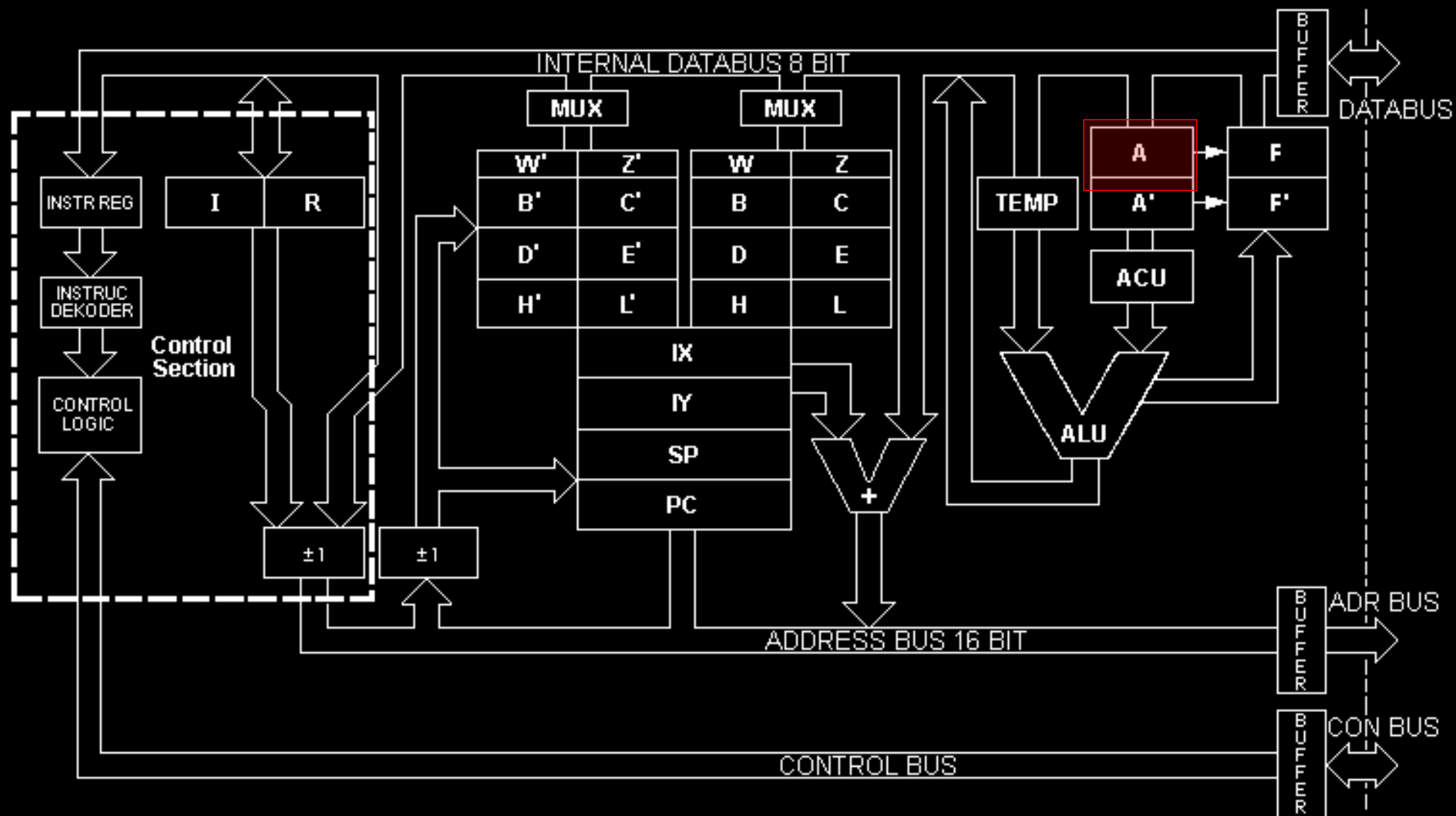


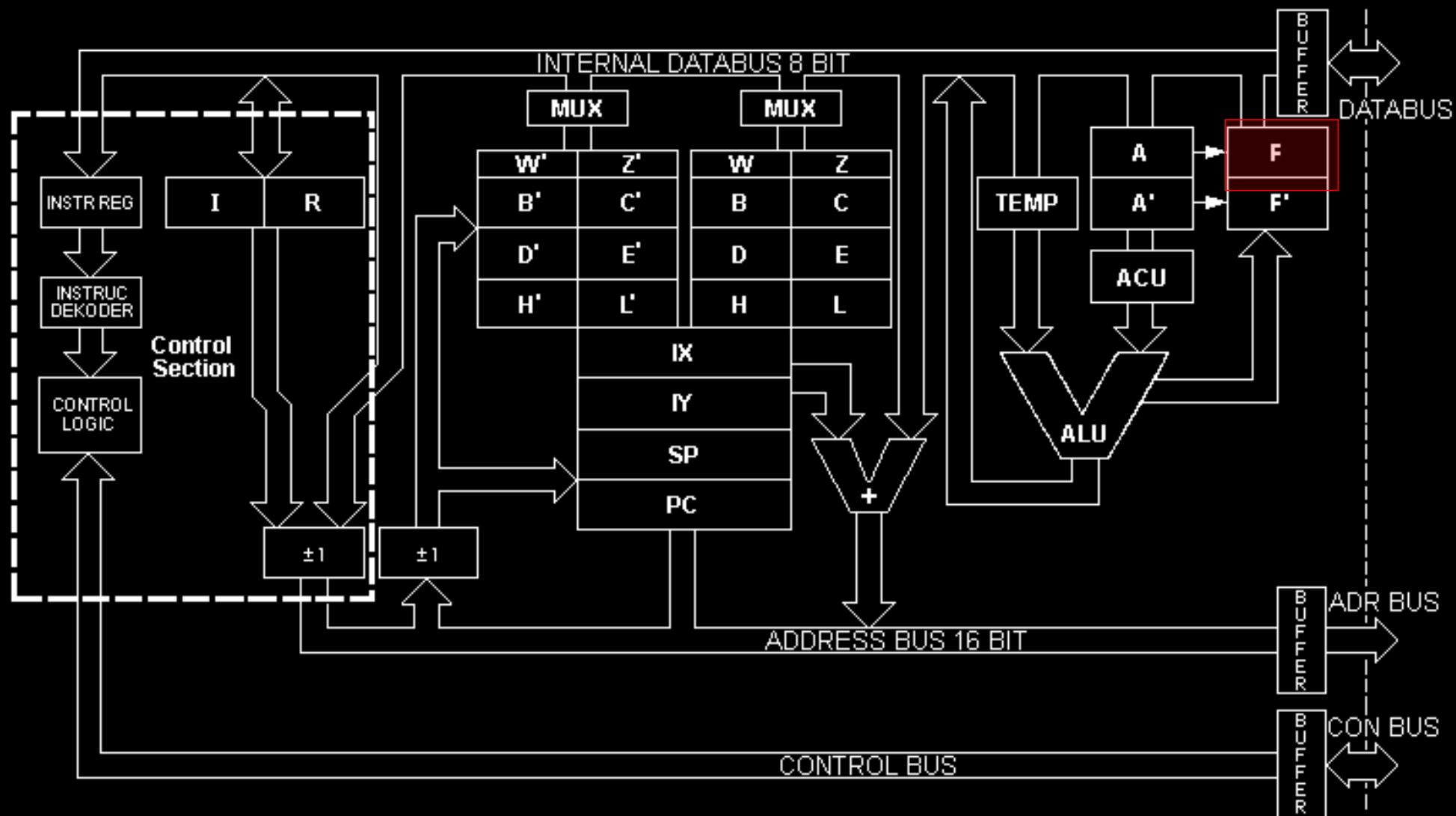


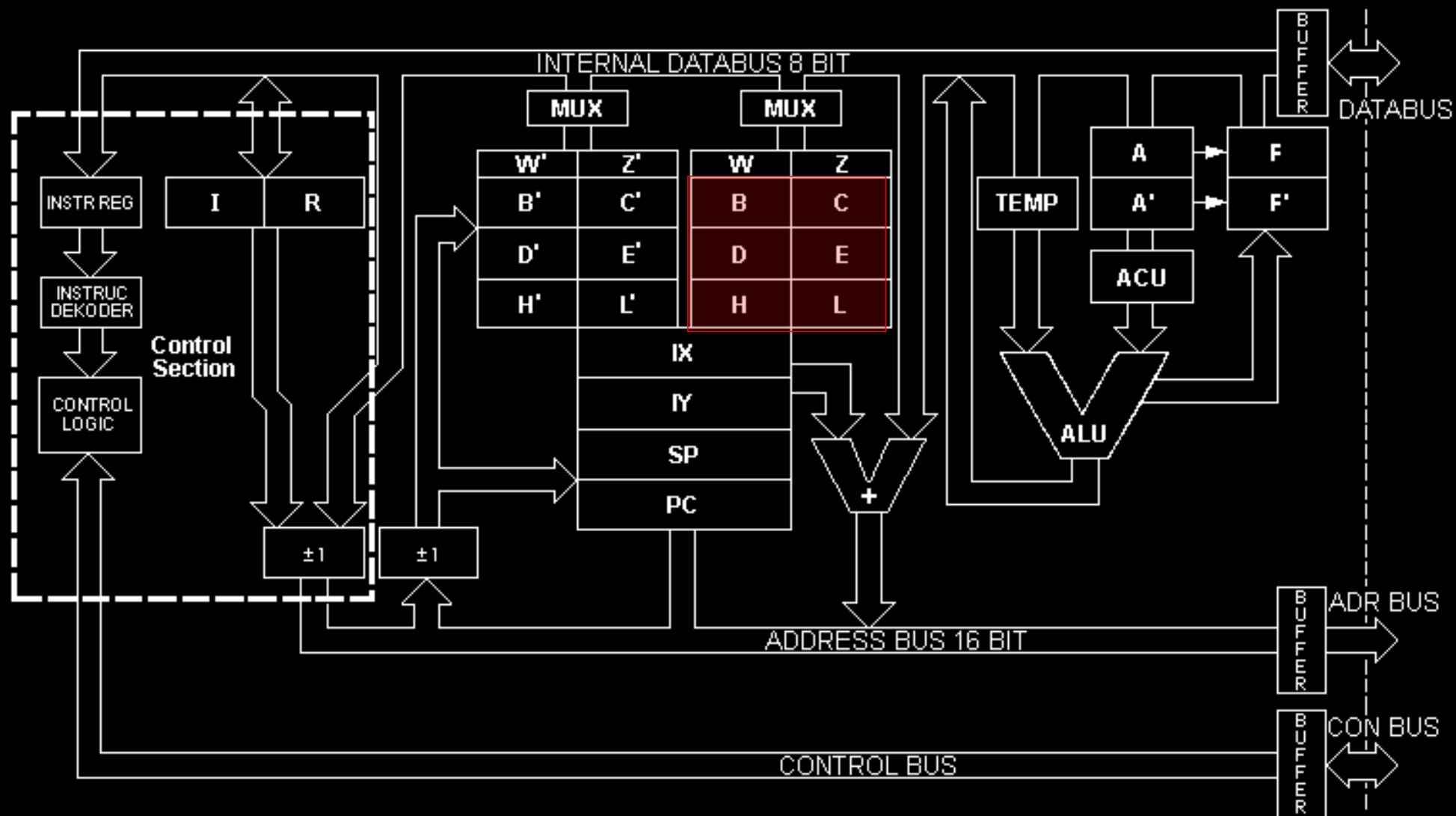


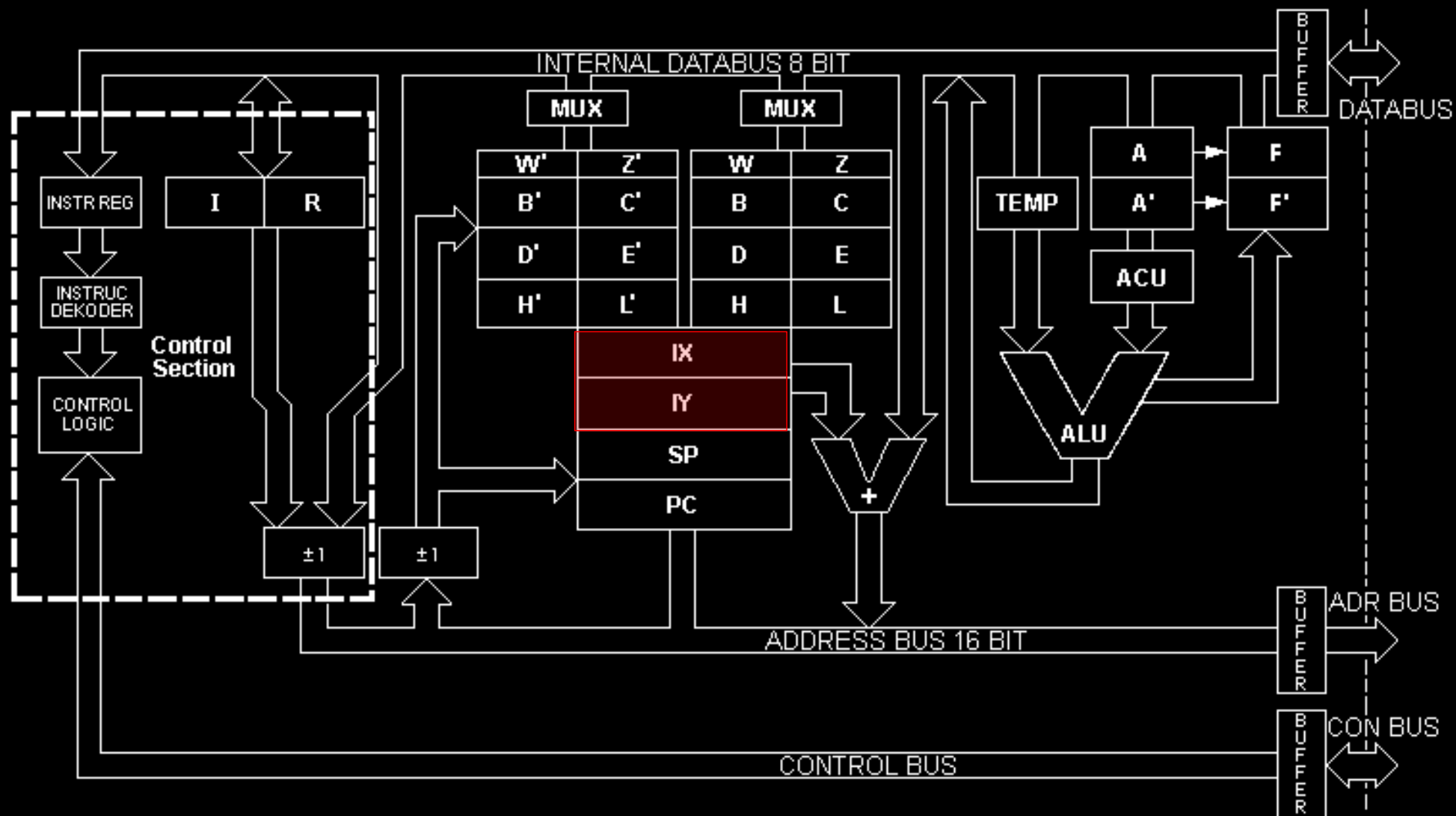


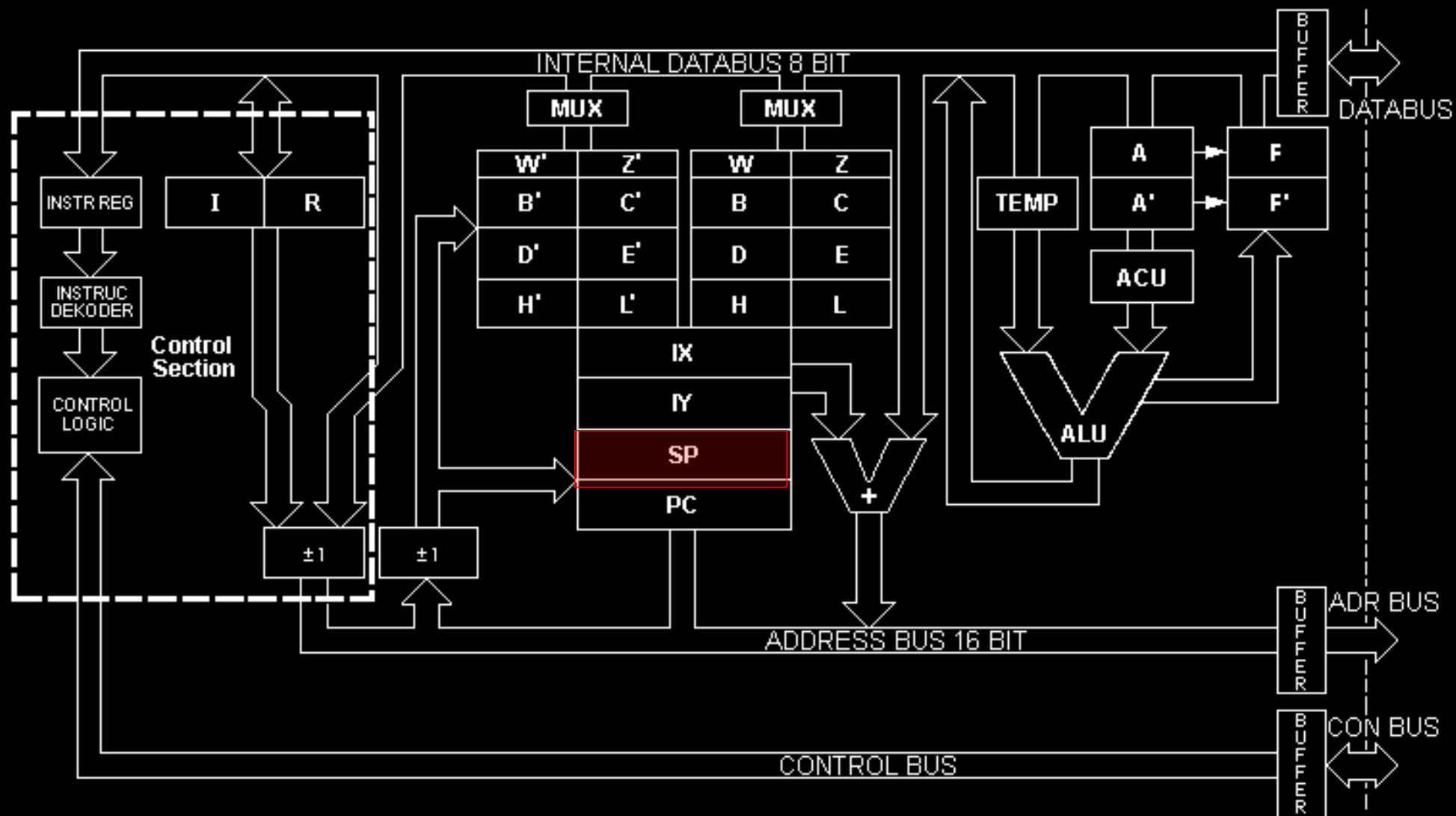


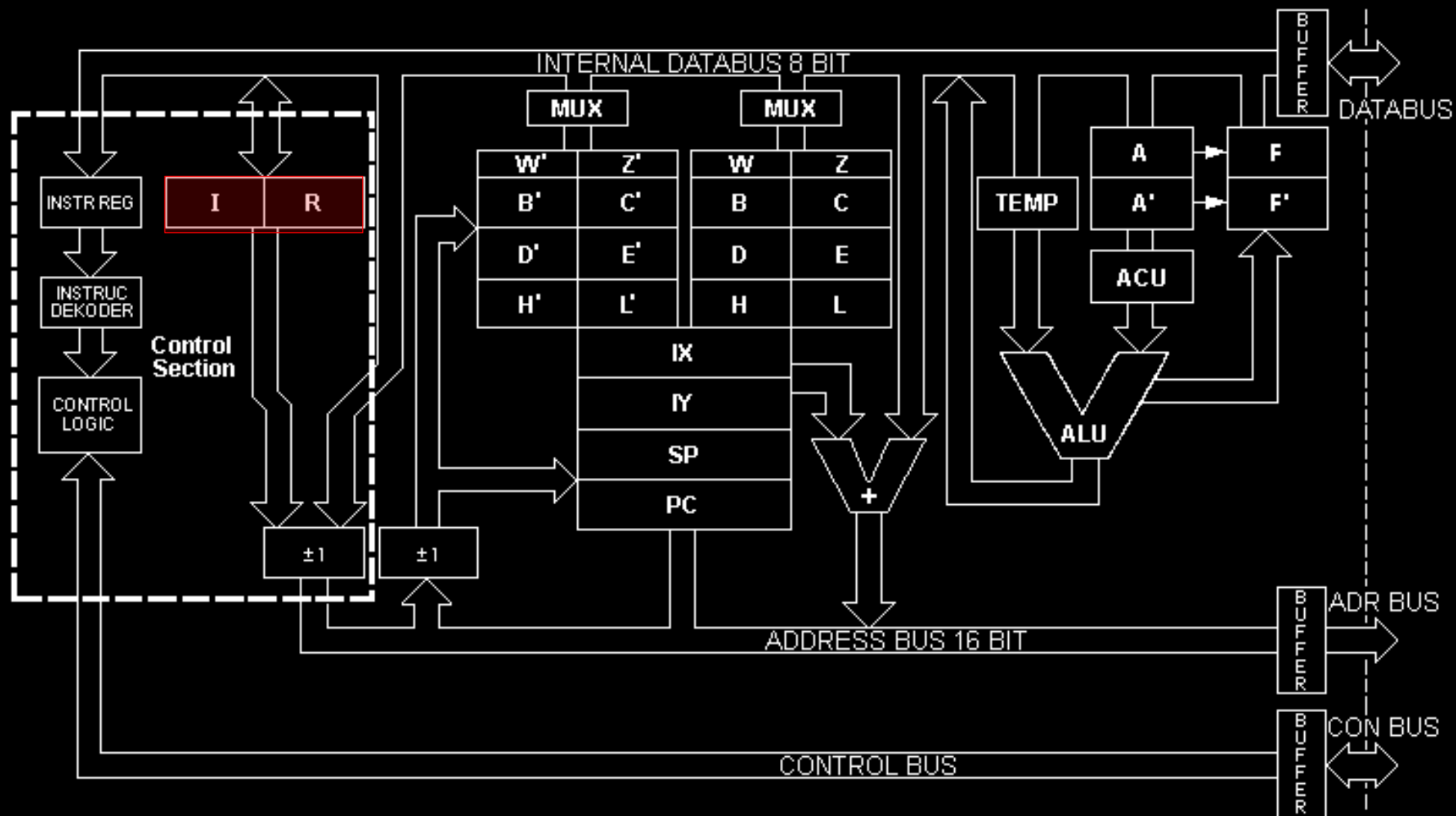


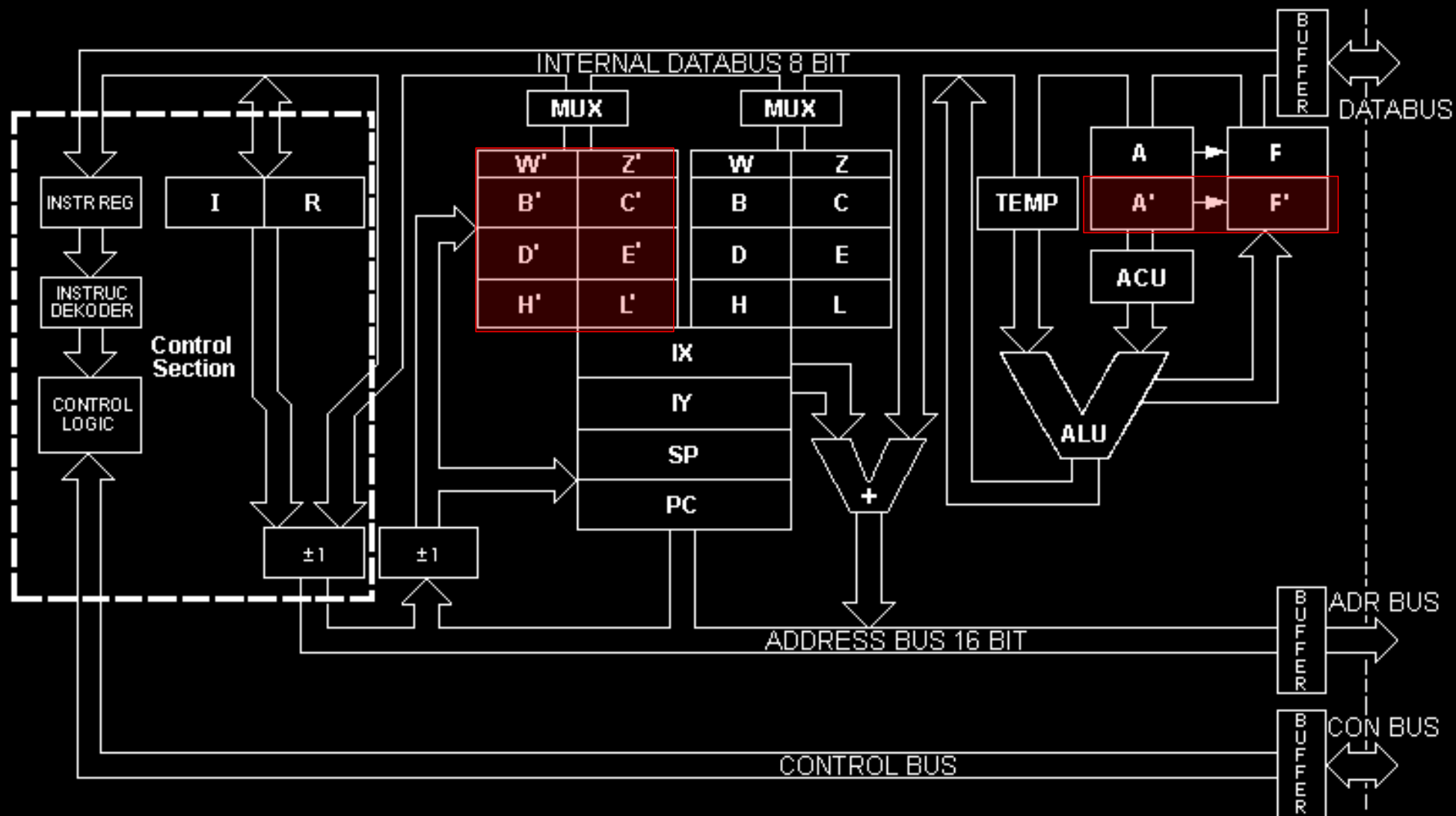


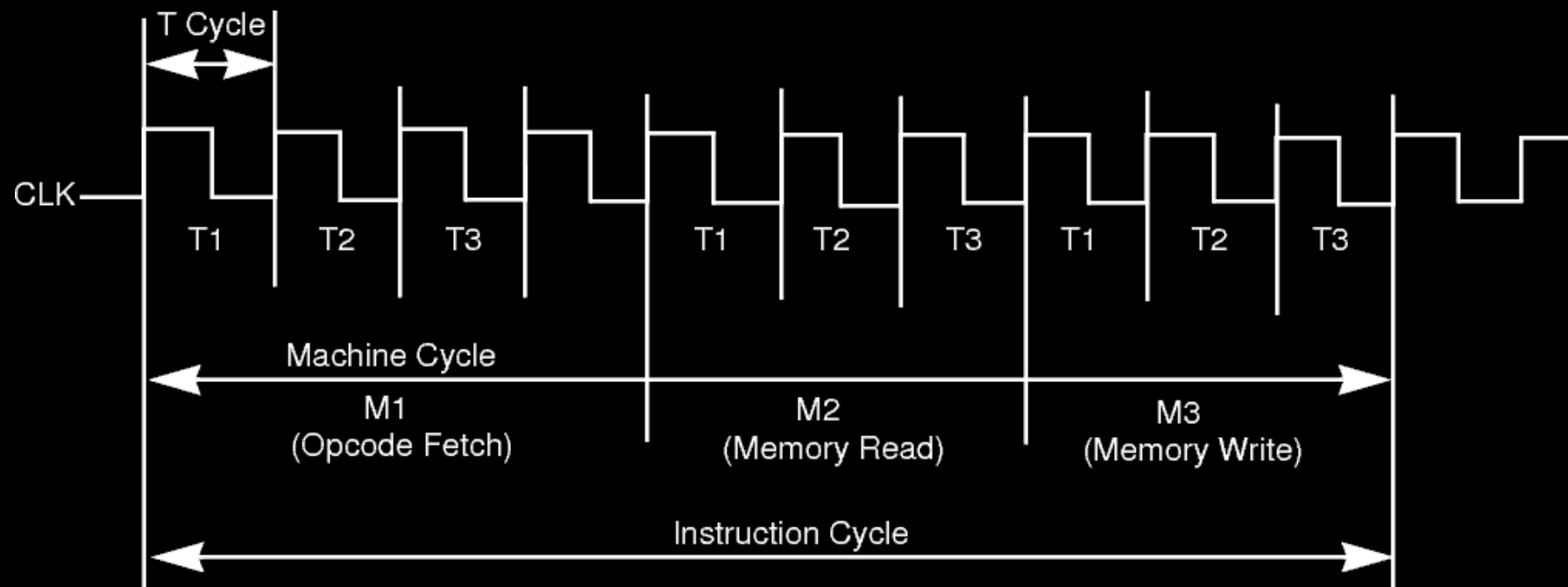


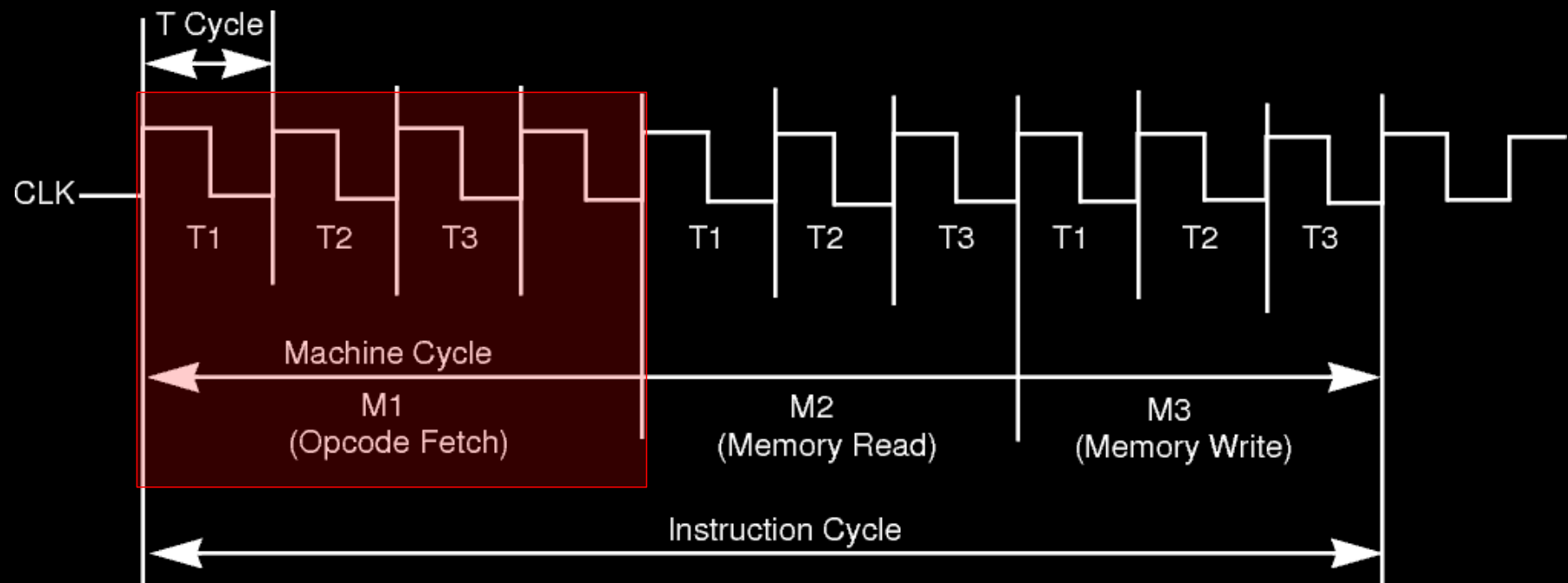


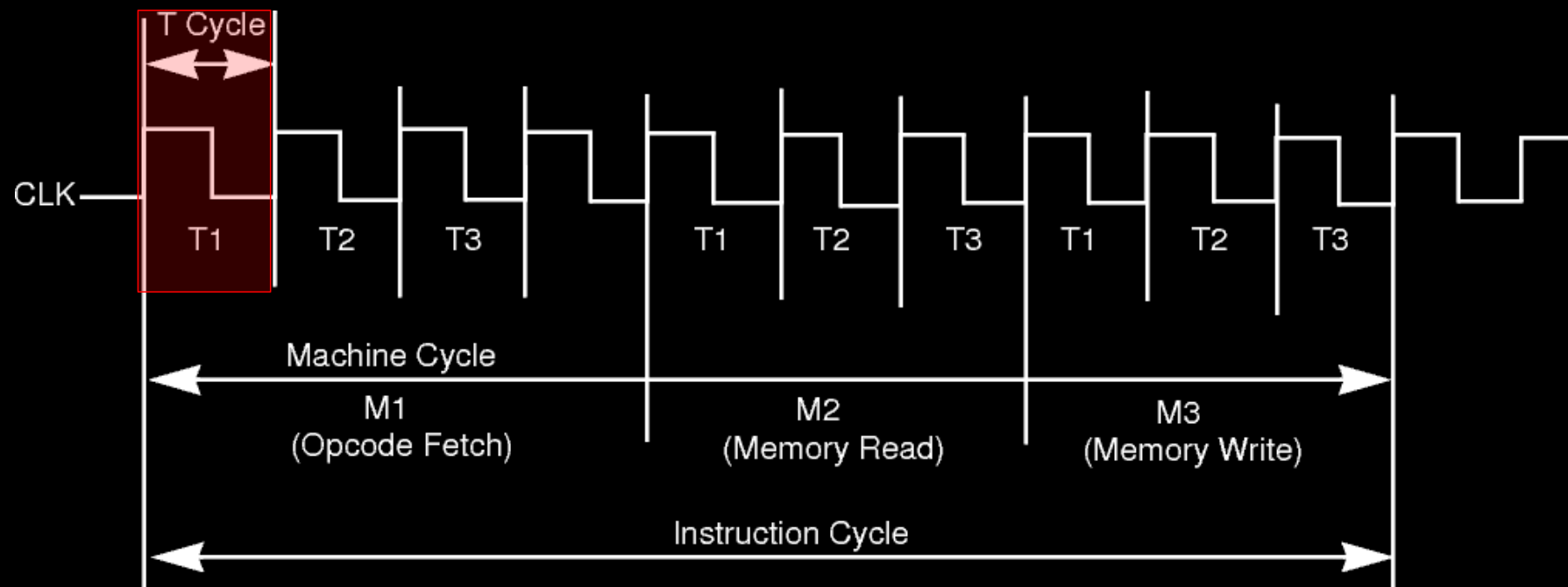


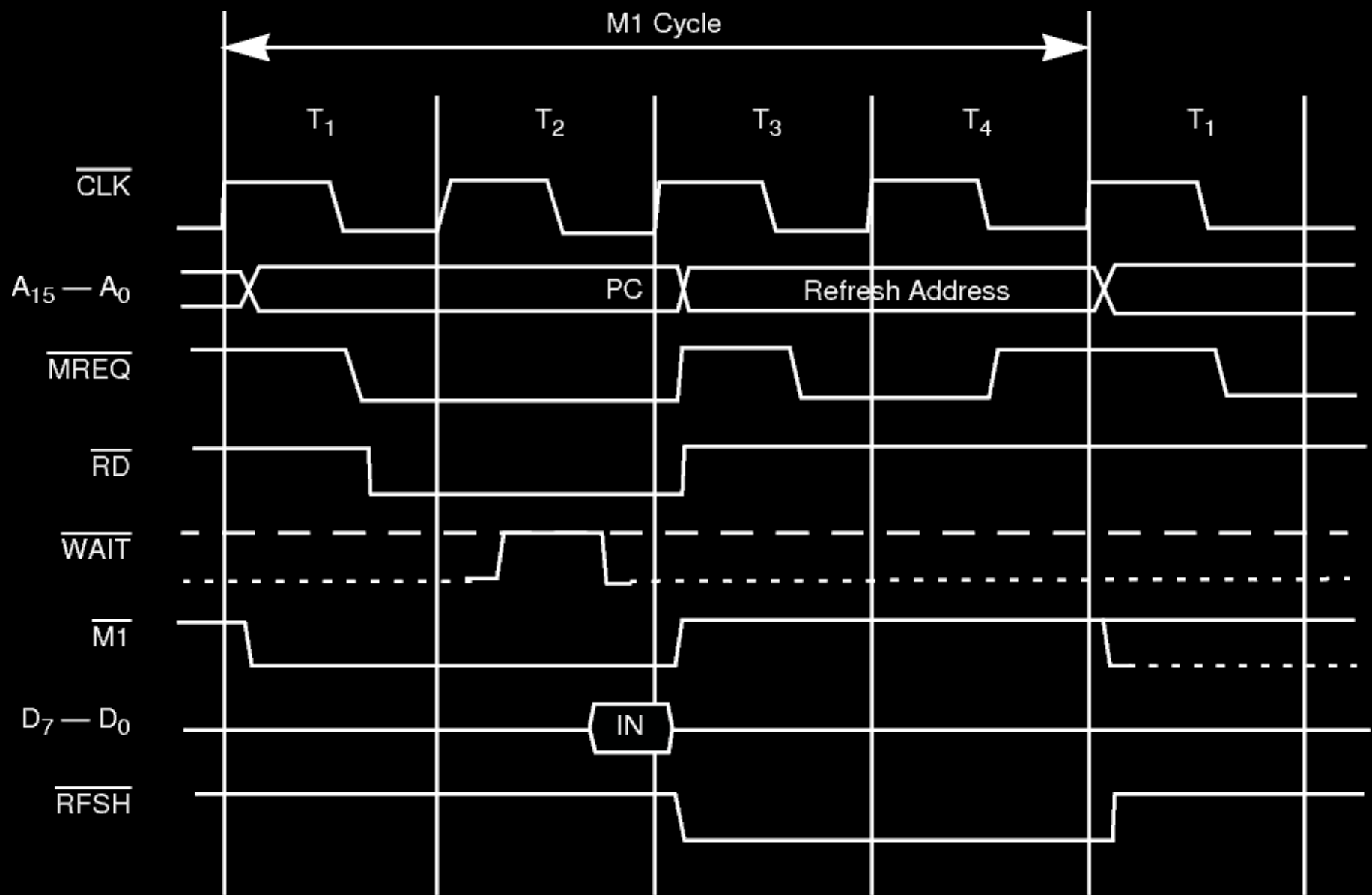


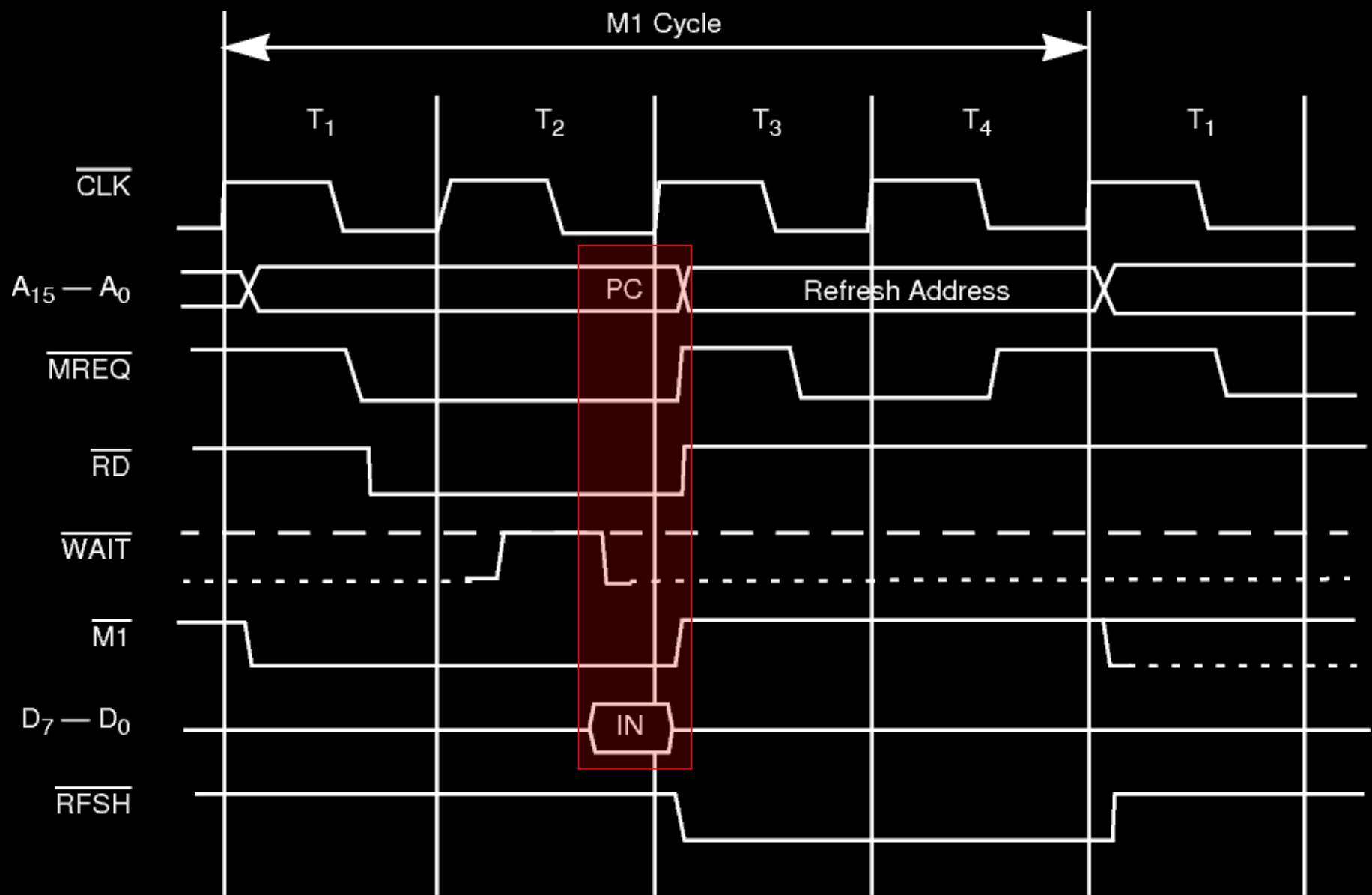


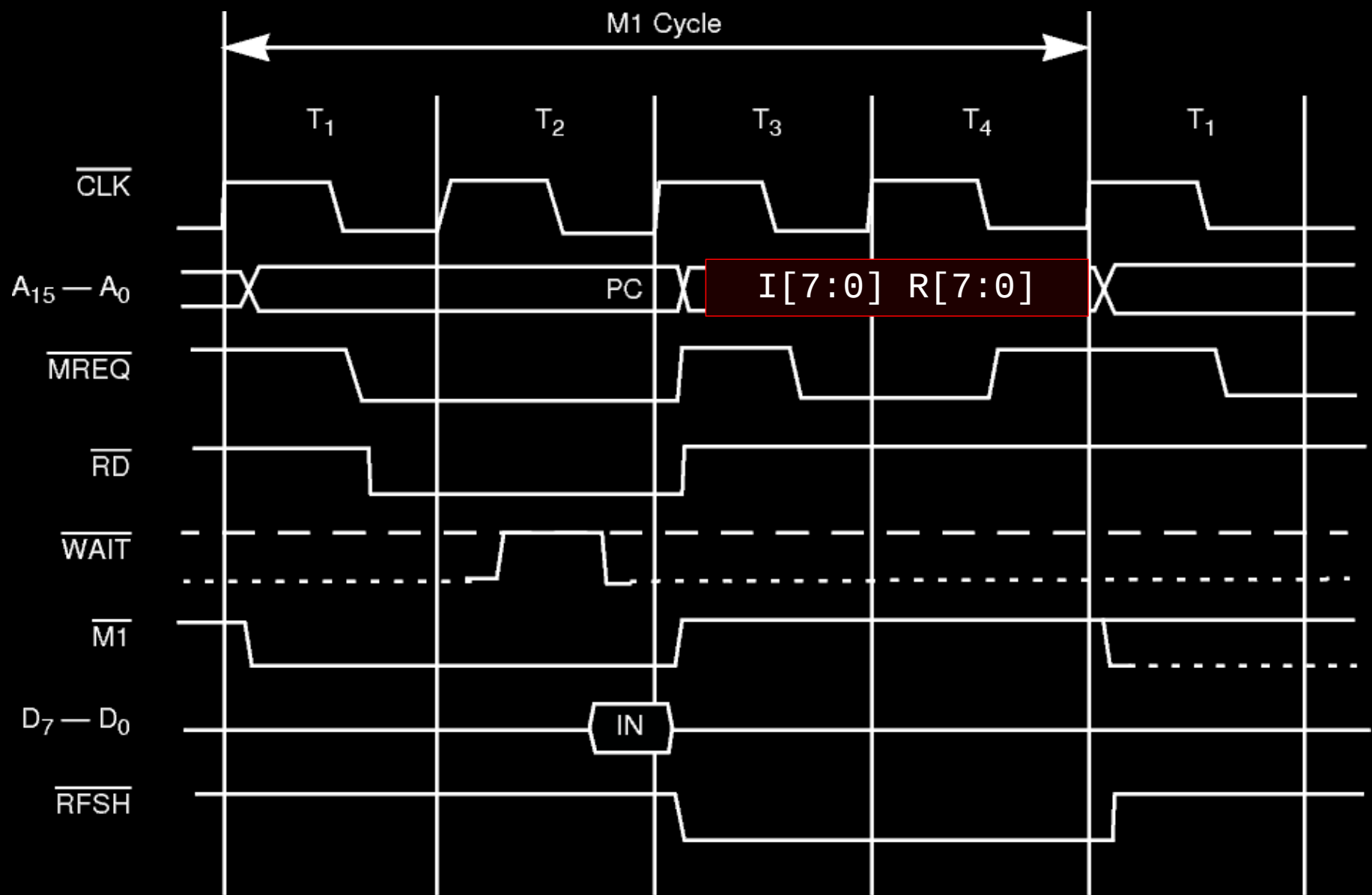


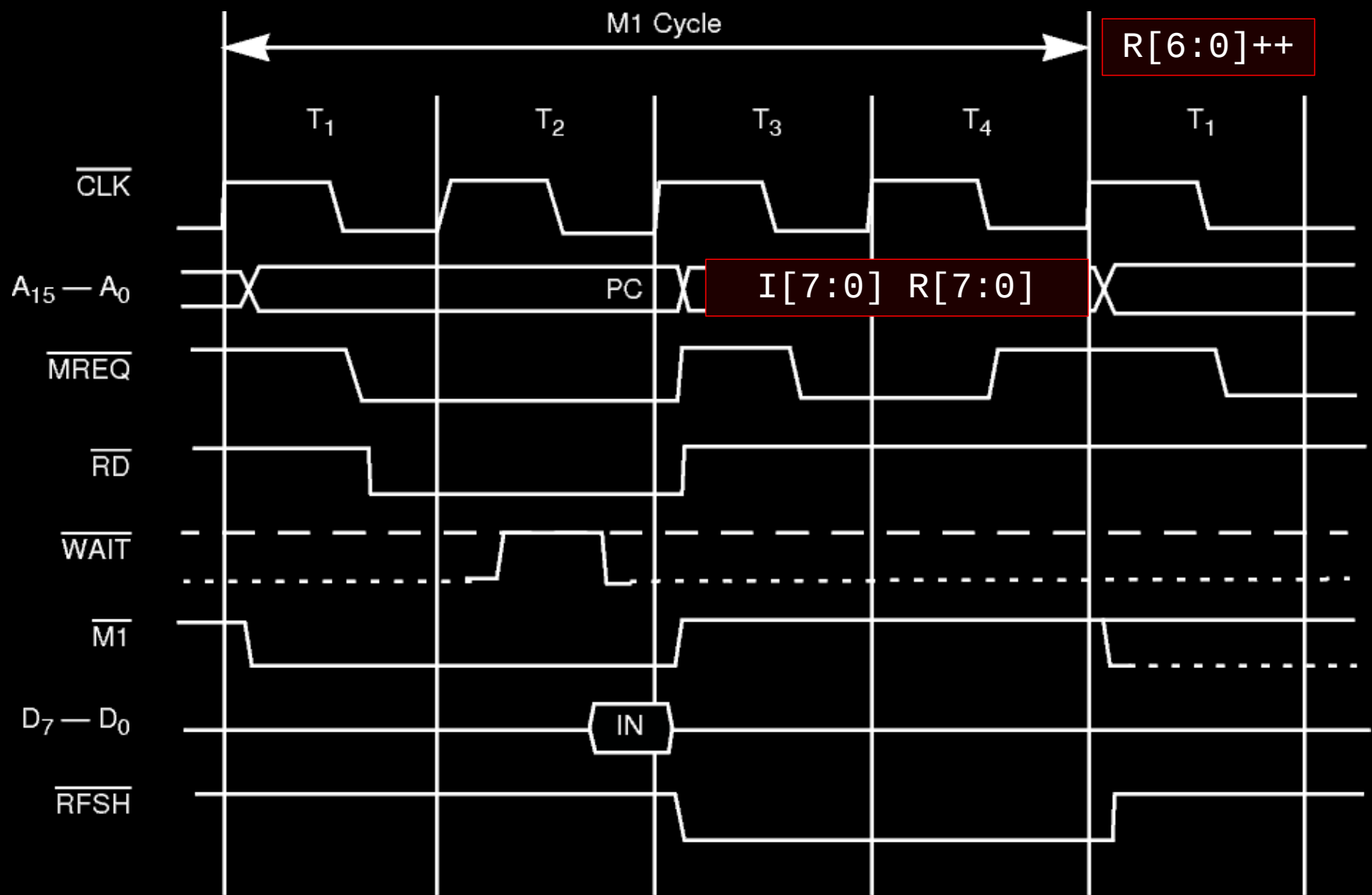








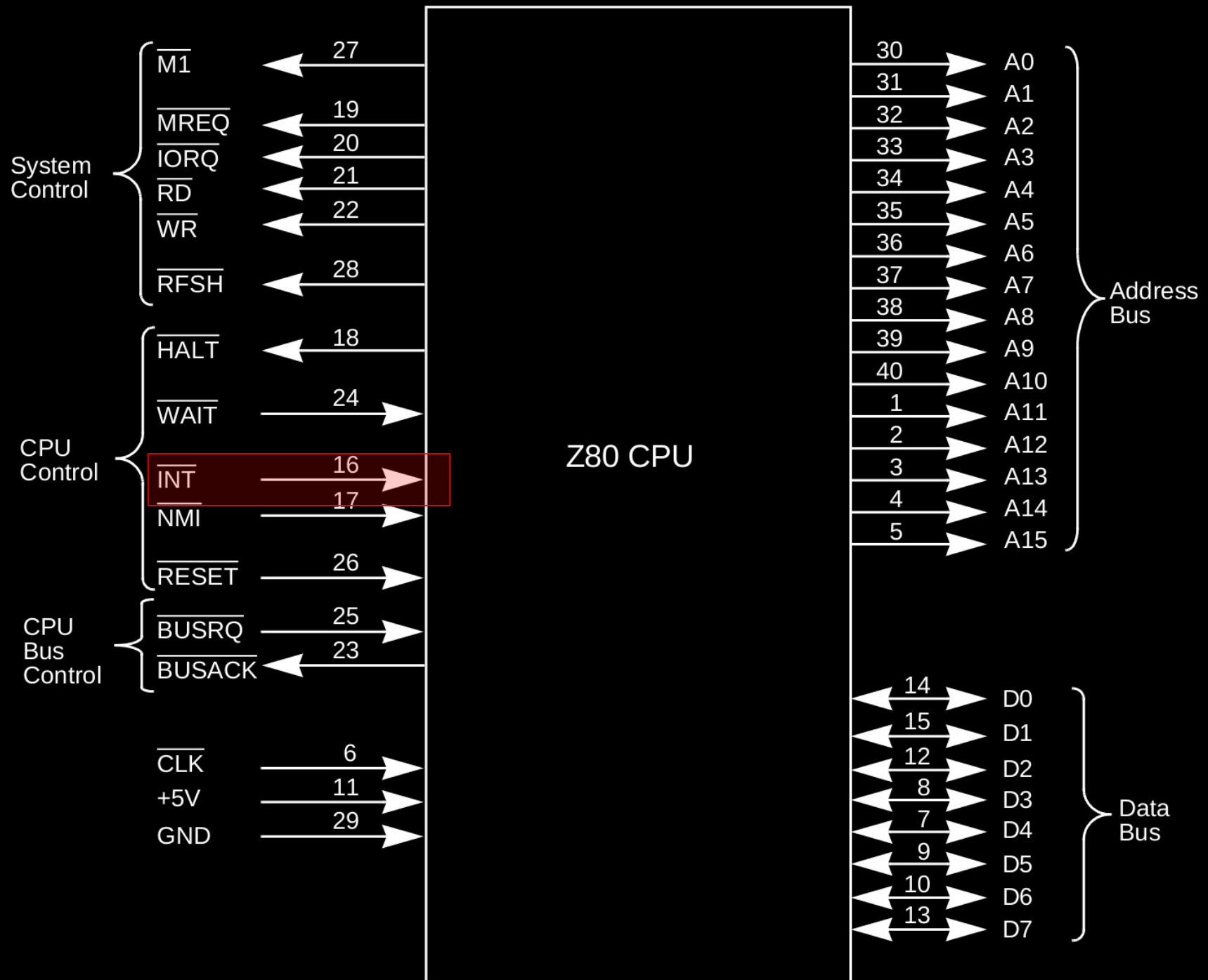




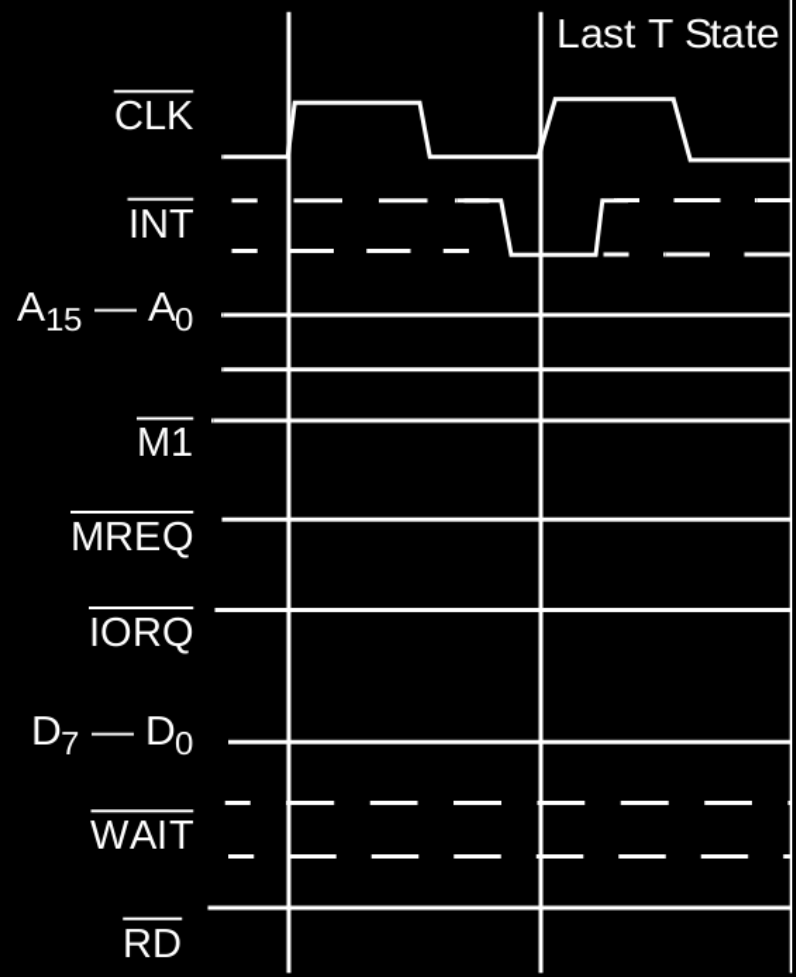
	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	nop	ld bc,nn	ld (bc),a	inc bc	inc b	dec b	ld b,n	rlca	ex af,af'	add hl,bc	ld a,(bc)	dec bc	inc c	dec c	ld c,n	rrca
10	djnz e	ld de,nn	ld (de),a	inc de	inc d	dec d	ld d,n	rla	jr e	add hl,de	ld a,(de)	dec de	inc e	dec e	ld e,n	rra
20	jr nz,e	ld hl,nn	ld (nn),hl	inc hl	inc h	dec h	ld h,n	daa	jr z,e	add hl,hl	ld hl,(nn)	dec hl	inc l	dec l	ld l,n	cpl
30	jr nc,e	ld sp,nn	ld (nn),a	inc sp	inc (hl)	dec (hl)	ld (hl),n	scf	jr c,e	add hl,sp	ld a,(nn)	dec sp	inc a	dec a	ld a,n	ccf
40	ld b,b	ld b,c	ld b,d	ld b,e	ld b,h	ld b,l	ld b,(hl)	ld b,a	ld c,b	ld c,c	ld c,d	ld c,e	ld c,h	ld c,l	ld c,(hl)	ld c,a
50	ld d,b	ld d,c	ld d,d	ld d,e	ld d,h	ld d,l	ld d,(hl)	ld d,a	ld e,b	ld e,c	ld e,d	ld e,e	ld e,h	ld e,l	ld e,(hl)	ld e,a
60	ld h,b	ld h,c	ld h,d	ld h,e	ld h,h	ld h,l	ld h,(hl)	ld h,a	ld l,b	ld l,c	ld l,d	ld l,e	ld l,h	ld l,l	ld l,(hl)	ld l,a
70	ld (hl),b	ld (hl),c	ld (hl),d	ld (hl),e	ld (hl),h	ld (hl),l	halt	ld (hl),a	ld a,b	ld a,c	ld a,d	ld a,e	ld a,h	ld a,l	ld a,(hl)	ld a,a
80	add a,b	add a,c	add a,d	add a,e	add a,h	add a,l	add a,(hl)	add a,a	adc a,b	adc a,c	adc a,d	adc a,e	adc a,h	adc a,l	adc a,(hl)	adc a,a
90	sub b	sub c	sub d	sub e	sub h	sub l	sub (hl)	sub a	sbc a,b	sbc a,c	sbc a,d	sbc a,e	sbc a,h	sbc a,l	sbc a,(hl)	sbc a,a
a0	and b	and c	and d	and e	and h	and l	and (hl)	and a	xor b	xor c	xor d	xor e	xor h	xor l	xor (hl)	xor a
b0	or b	or c	or d	or e	or h	or l	or (hl)	or a	cp b	cp c	cp d	cp e	cp h	cp l	cp (hl)	cp a
c0	ret nz	pop bc	jp nz,nn	jp nn	call nz,nn	push bc	add a,n	rst 0	ret z	ret	jp z,nn		call z,nn	call nn	adc a,n	rst 08h
d0	ret nc	pop de	jp nc,nn	out (n),a	call nc,nn	push de	sub n	rst 10h	ret c	exx	jp c,nn	in a,(n)	call c,nn		sbc a,n	rst 18h
e0	ret po	pop hl	jp po,nn	ex (sp),hl	call po,nn	push hl	and n	rst 20h	ret pe	jp (hl)	jp pe,nn	ex de,hl	call pe,nn		xor n	rst 28h
f0	ret p	pop af	jp p,nn	di	call p,nn	push af	or n	rst 30h	ret m	ld sp,hl	jp m,nn	ei	call m,nn		cp n	rst 38h

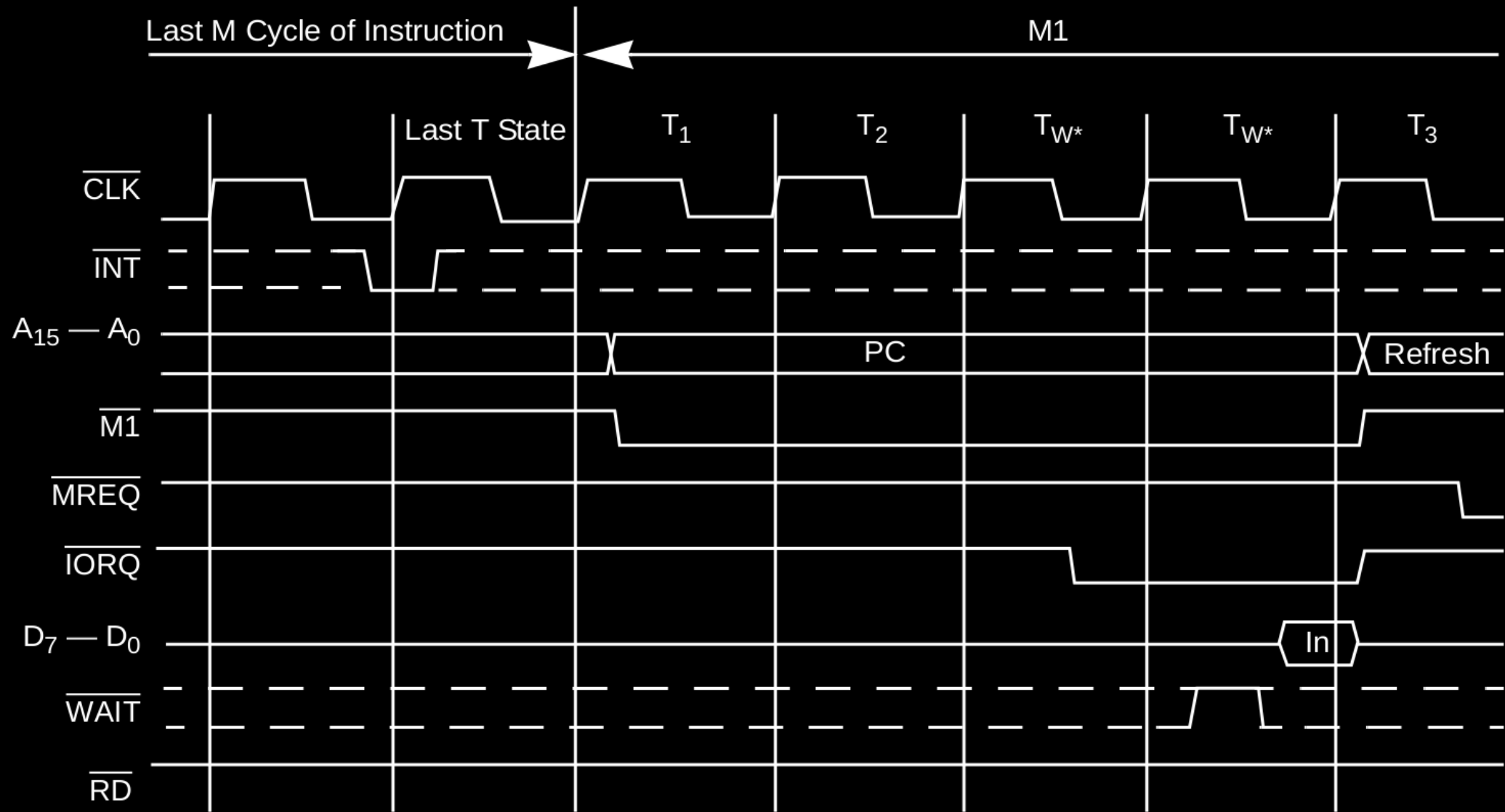
	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f	
00	nop	ld bc,nn	ld (bc),a	inc bc	inc b	dec b	ld b,n	rlca	ex af,af'	add hl,bc	ld a,(bc)	dec bc	inc c	dec c	ld c,n	rrca	
10	djnz e	ld de,nn	ld (de),a	inc de	inc d	dec d	ld d,n	rla	jr e	add hl,de	ld a,(de)	dec de	inc e	dec e	ld e,n	rra	
20	jr nz,e	ld hl,nn	ld (nn),hl	inc hl	inc h	dec h	ld h,n	daa	jr z,e	add hl,hl	ld hl,(nn)	dec hl	inc l	dec l	ld l,n	cpl	
30	jr nc,e	ld sp,nn	ld (nn),a	inc sp	inc (hl)	dec (hl)	ld (hl),n	scf	jr c,e	add hl,sp	ld a,(nn)	dec sp	inc a	dec a	ld a,n	ccf	
40	ld b,b	ld b,c	ld b,d	ld b,e	ld b,h	ld b,l	ld b,(hl)	ld b,a	ld c,b	ld c,c	ld c,d	ld c,e	ld c,h	ld c,l	ld c,(hl)	ld c,a	
50	ld d,b	ld d,c	ld d,d	ld d,e	ld d,h	ld d,l	ld d,(hl)	ld d,a	ld e,b	ld e,c	ld e,d	ld e,e	ld e,h	ld e,l	ld e,(hl)	ld e,a	
60	ld h,b	ld h,c	ld h,d	ld h,e	ld h,h	ld h,l	ld h,(hl)	ld h,a	ld l,b	ld l,c	ld l,d	ld l,e	ld l,h	ld l,l	ld l,(hl)	ld l,a	
70	ld (hl),b	ld (hl),c	ld (hl),d	ld (hl),e	ld (hl),h	ld (hl),l	halt	ld (hl),a	ld a,b	ld a,c	ld a,d	ld a,e	ld a,h	ld a,l	ld a,(hl)	ld a,a	
80	add a,b	add a,c	add a,d	add a,e	add a,h	add a,l	add a,(hl)	add a,a	adc a,b	adc a,c	adc a,d	adc a,e	adc a,h	adc a,l	adc a,(hl)	adc a,a	
90	sub b	sub c	sub d	sub e	sub h	sub l	sub (hl)	sub a	sbc a,b	sbc a,c	sbc a,d	sbc a,e	sbc a,h	sbc a,l	sbc a,(hl)	sbc a,a	
a0	and b	and c	and d	and e	and h	and l	and (hl)	and a	xor b	xor c	xor d	xor e	xor h	xor l	xor (hl)	xor a	
b0	or b	or c	or d	or e	or h	or l	or (hl)	or a	cp b	cp c	cp d	cp e	cp h	cp l	cp (hl)	cp a	
c0	ret nz	pop bc	jp nz,nn	jp nn	call nz,nn	push bc	add a,n	rst 0	ret z	ret	jp z,nn	cb xx		ld n	call nn	adc a,n	rst 08h
d0	ret nc	pop de	jp nc,nn	out (n),a	call nc,nn	push de	sub n	rst 10h	ret c	exx	jp c,nn	in a,(n)	call c,nn	dd xx		a,n	rst 18h
e0	ret po	pop hl	jp po,nn	ex (sp),hl	call po,nn	push hl	and n	rst 20h	ret pe	jp (hl)	jp pe,nn	ex de,hl	call pe,nn	ed xx		n	rst 28h
f0	ret p	pop af	jp p,nn	di	call p,nn	push af	or n	rst 30h	ret m	ld sp,hl	jp m,nn	ei	call m,nn	fd xx			rst 38h

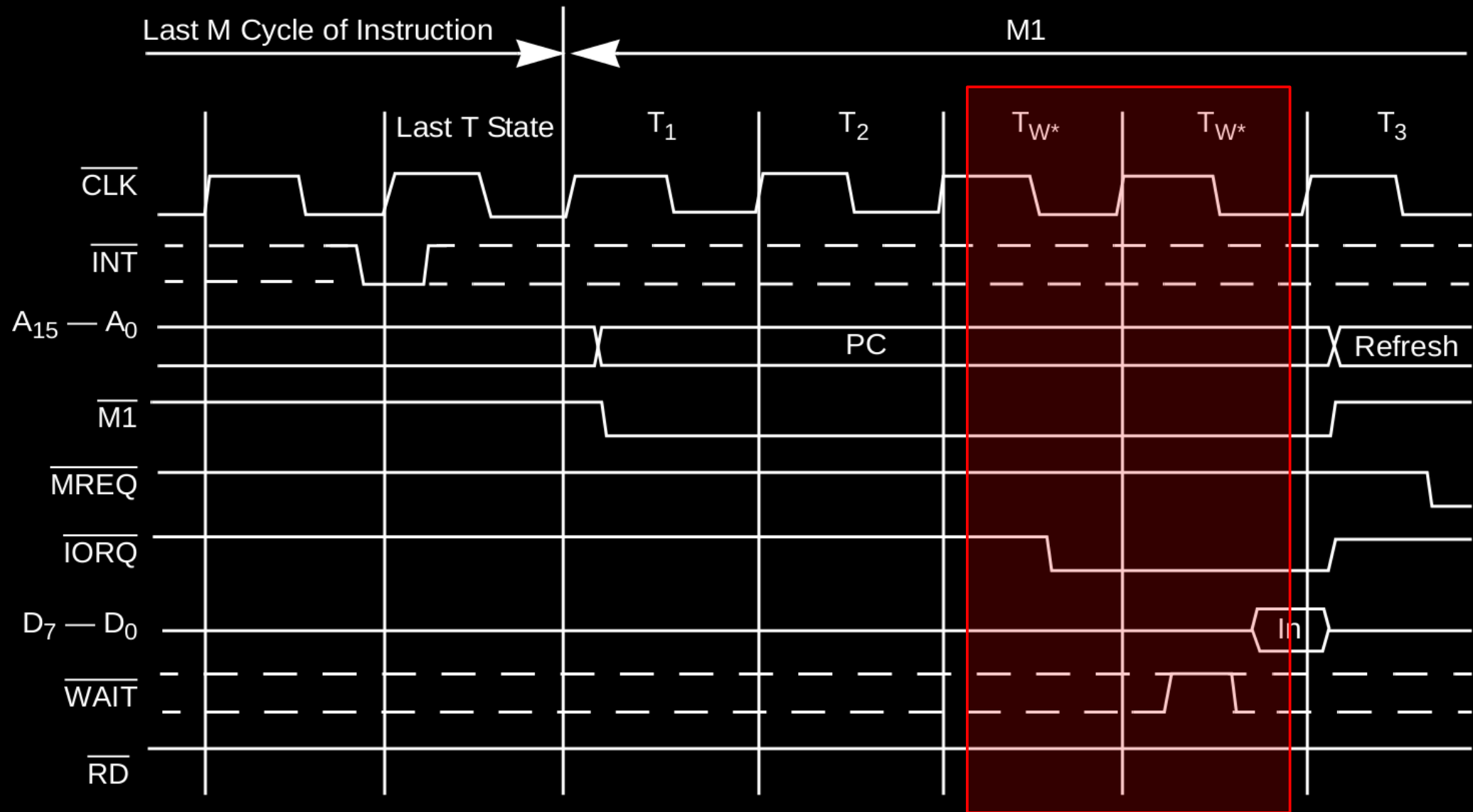
	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	nop	ld bc,nn	ld (bc),a	inc bc	inc b	dec b	ld b,n	rlca	ex af,af'	add hl,bc	ld a,(bc)	dec bc	inc c	dec c	ld c,n	rrca
10	djnz e	ld de,nn	ld (de),a	inc de	inc d	dec d	ld d,n	rla	jr e	add hl,de	ld a,(de)	dec de	inc e	dec e	ld e,n	rra
20	jr nz,e	ld hl,nn	ld (nn),hl	inc hl	inc h	dec h	ld h,n	daa	jr z,e	add hl,hl	ld hl,(nn)	dec hl	inc l	dec l	ld l,n	cpl
30	jr nc,e	ld sp,nn	ld (nn),a	inc sp	inc (hl)	dec (hl)	ld (hl),n	scf	jr c,e	add hl,sp	ld a,(nn)	dec sp	inc a	dec a	ld a,n	ccf
40	ld b,b	ld b,c	ld b,d	ld b,e	ld b,h	ld b,l	ld b,(hl)	ld b,a	ld c,b	ld c,c	ld c,d	ld c,e	ld c,h	ld c,l	ld c,(hl)	ld c,a
50	ld d,b	ld d,c	ld d,d	ld d,e	ld d,h	ld d,l	ld d,(hl)	ld d,a	ld e,b	ld e,c	ld e,d	ld e,e	ld e,h	ld e,l	ld e,(hl)	ld e,a
60	ld h,b	ld h,c	ld h,d	ld h,e	ld h,h	ld h,l	ld h,(hl)	ld h,a	ld l,b	ld l,c	ld l,d	ld l,e	ld l,h	ld l,l	ld l,(hl)	ld l,a
70	ld (hl),b	ld (hl),c	ld (hl),d	ld (hl),e	ld (hl),h	ld (hl),l	halt	ld (hl),a	ld a,b	ld a,c	ld a,d	ld a,e	ld a,h	ld a,l	ld a,(hl)	ld a,a
80	add a,b	add a,c	add a,d	add a,e	add a,h	add a,l	add a,(hl)	add a,a	adc a,b	adc a,c	adc a,d	adc a,e	adc a,h	adc a,l	adc a,(hl)	adc a,a
90	sub b	sub c	sub d	sub e	sub h	sub l	sub (hl)	sub a	sbc a,b	sbc a,c	sbc a,d	sbc a,e	sbc a,h	sbc a,l	sbc a,(hl)	sbc a,a
a0	and b	and c	and d	and e	and h	and l	and (hl)	and a	xor b	xor c	xor d	xor e	xor h	xor l	xor (hl)	xor a
b0	or b	or c	or d	or e	or h	or l	or (hl)	or a	cp b	cp c	cp d	cp e	cp h	cp l	cp (hl)	cp a
c0	ret nz	pop bc	jp nz,nn	jp nn	call nz,nn	push bc	add a,n	rst 0	ret z	ret	jp z,nn	cb xx	ld n	call nn	adc a,n	rst 08h
d0	ret nc	pop de	jp nc,nn	out (n),a	call nc,nn	push de	sub n	rst 10h	ret c	exx	jp c,nn	in a,(n)	call c,nn	dd cb xx		18h
e0	ret po	pop hl	jp po,nn	ex (sp),hl	call po,nn	push hl	and n	rst 20h	ret pe	jp (hl)	jp pe,nn	ex de,hl	call pe,nn	ed cb xx		28h
f0	ret p	pop af	jp p,nn	di	call p,nn	push af	or n	rst 30h	ret m	ld sp,hl	jp m,nn	ei	call m,nn	fd xx		rst 38h

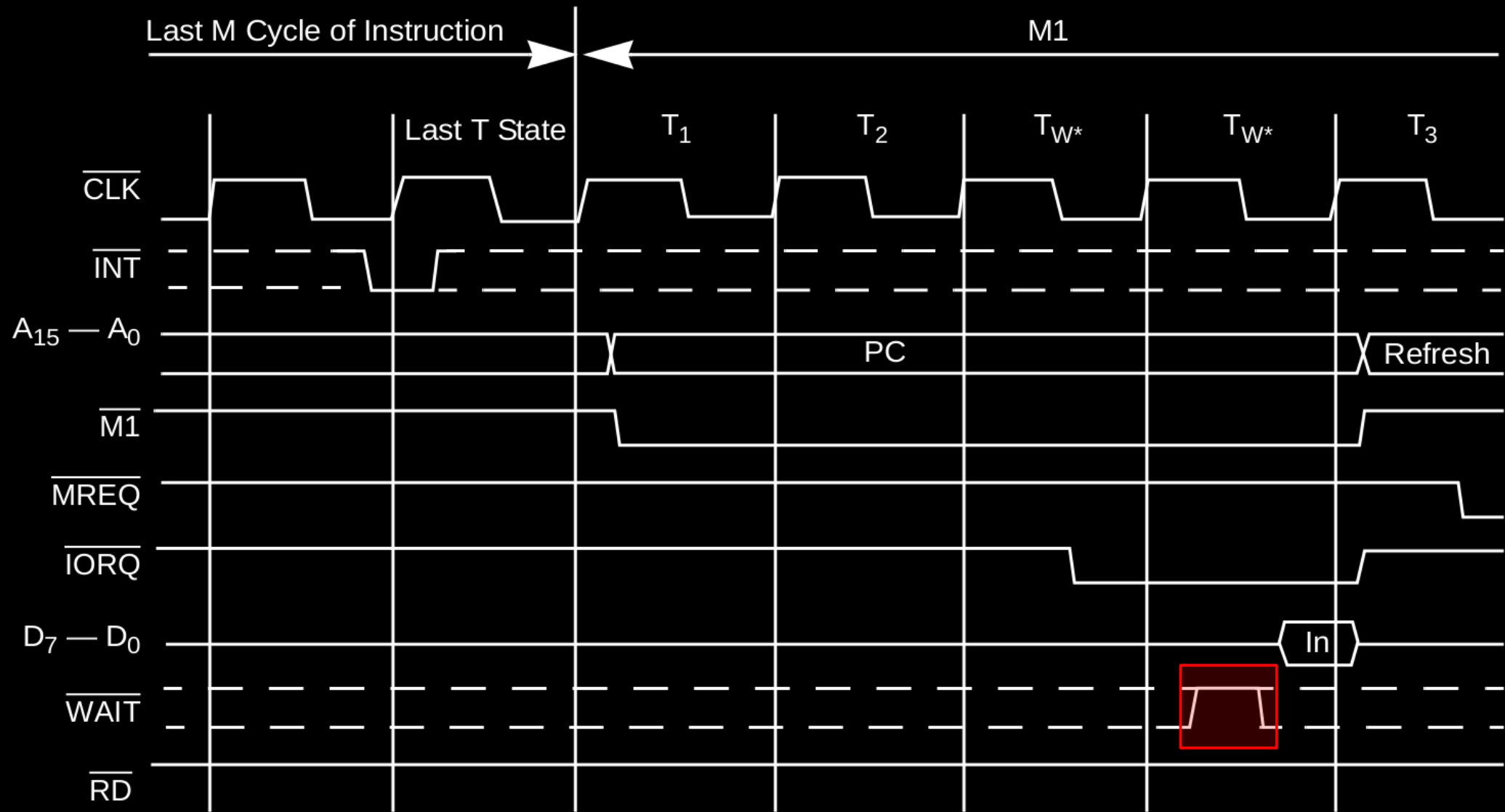


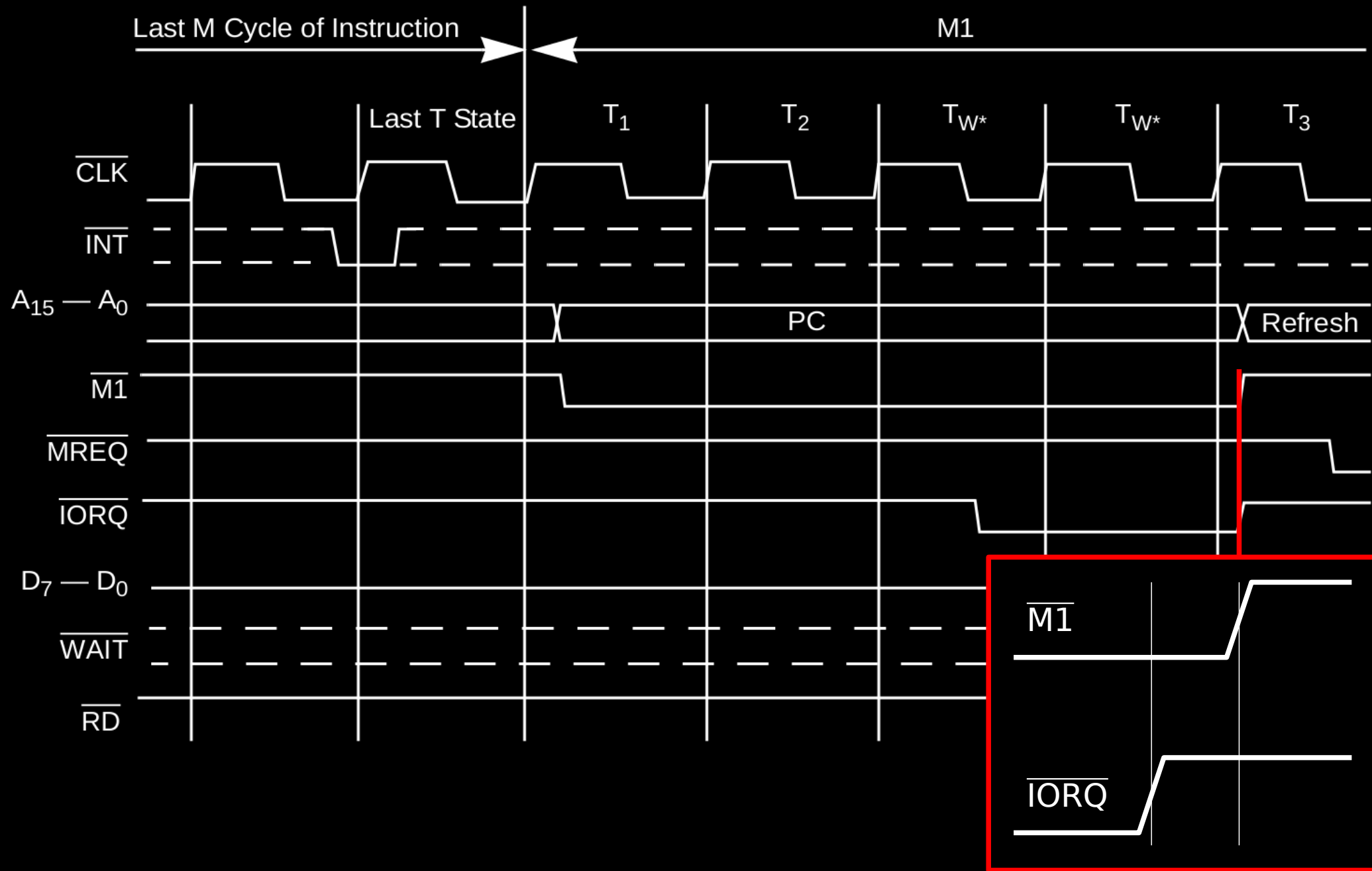
Last M Cycle of Instruction →

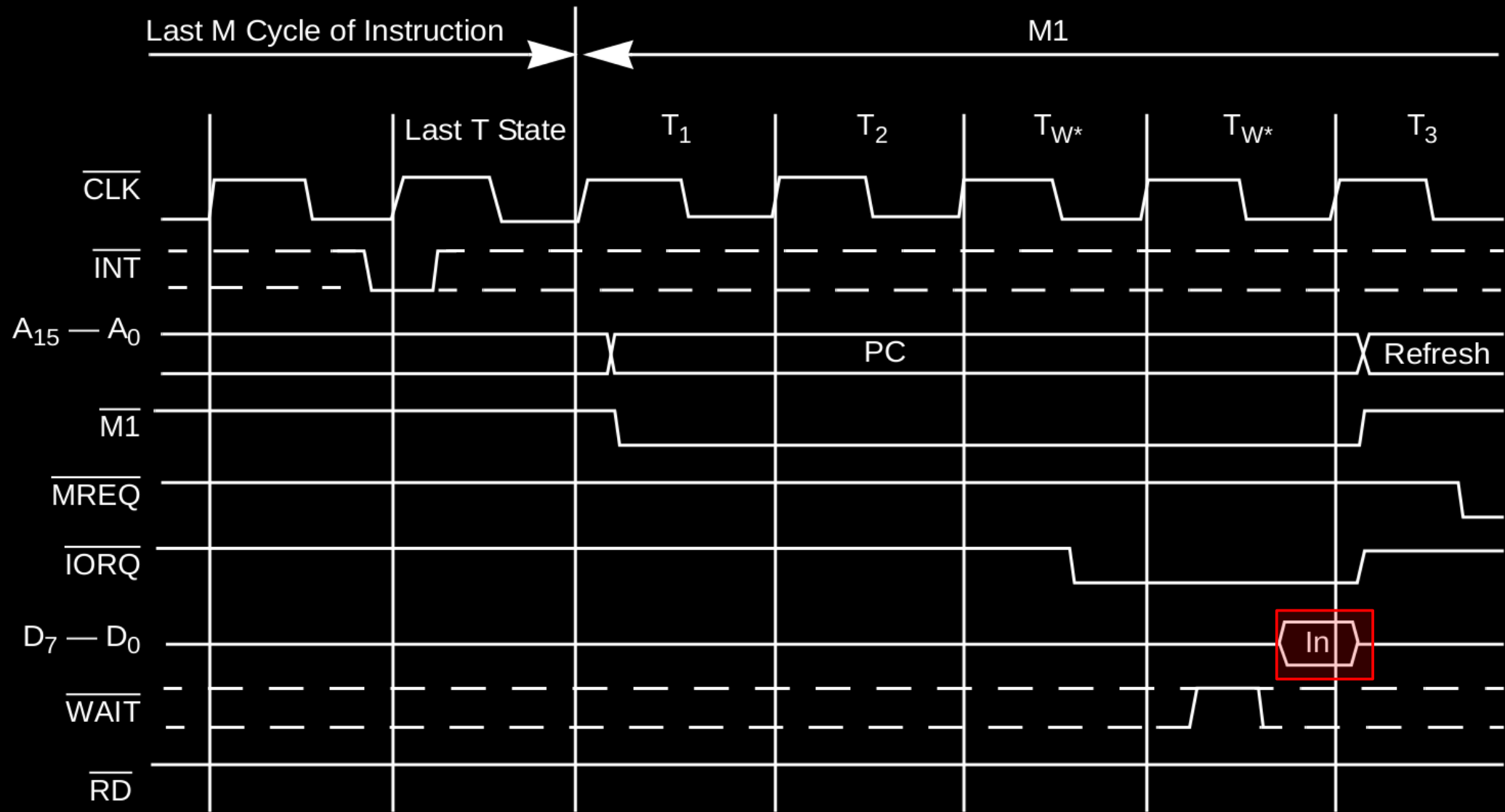


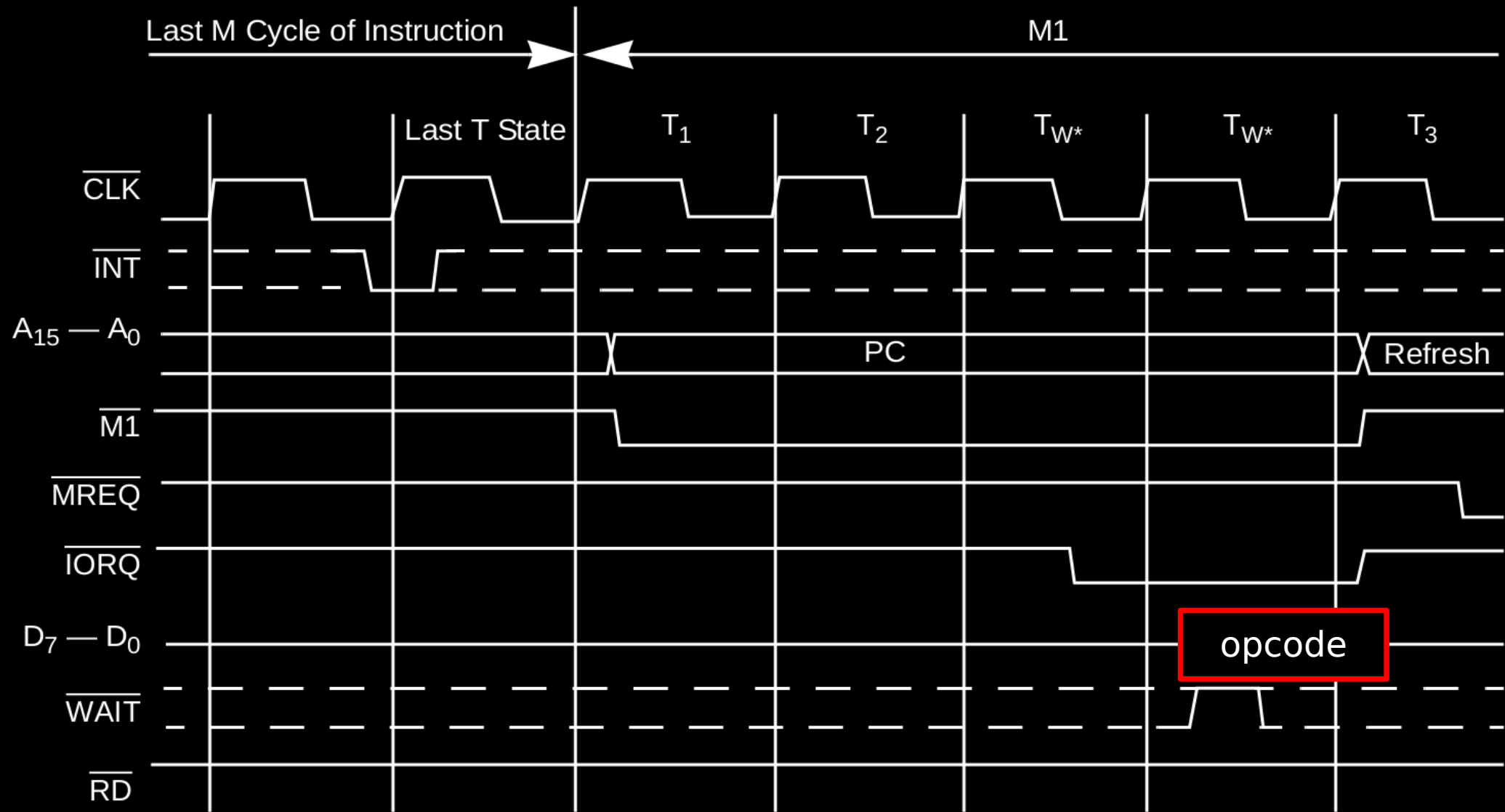






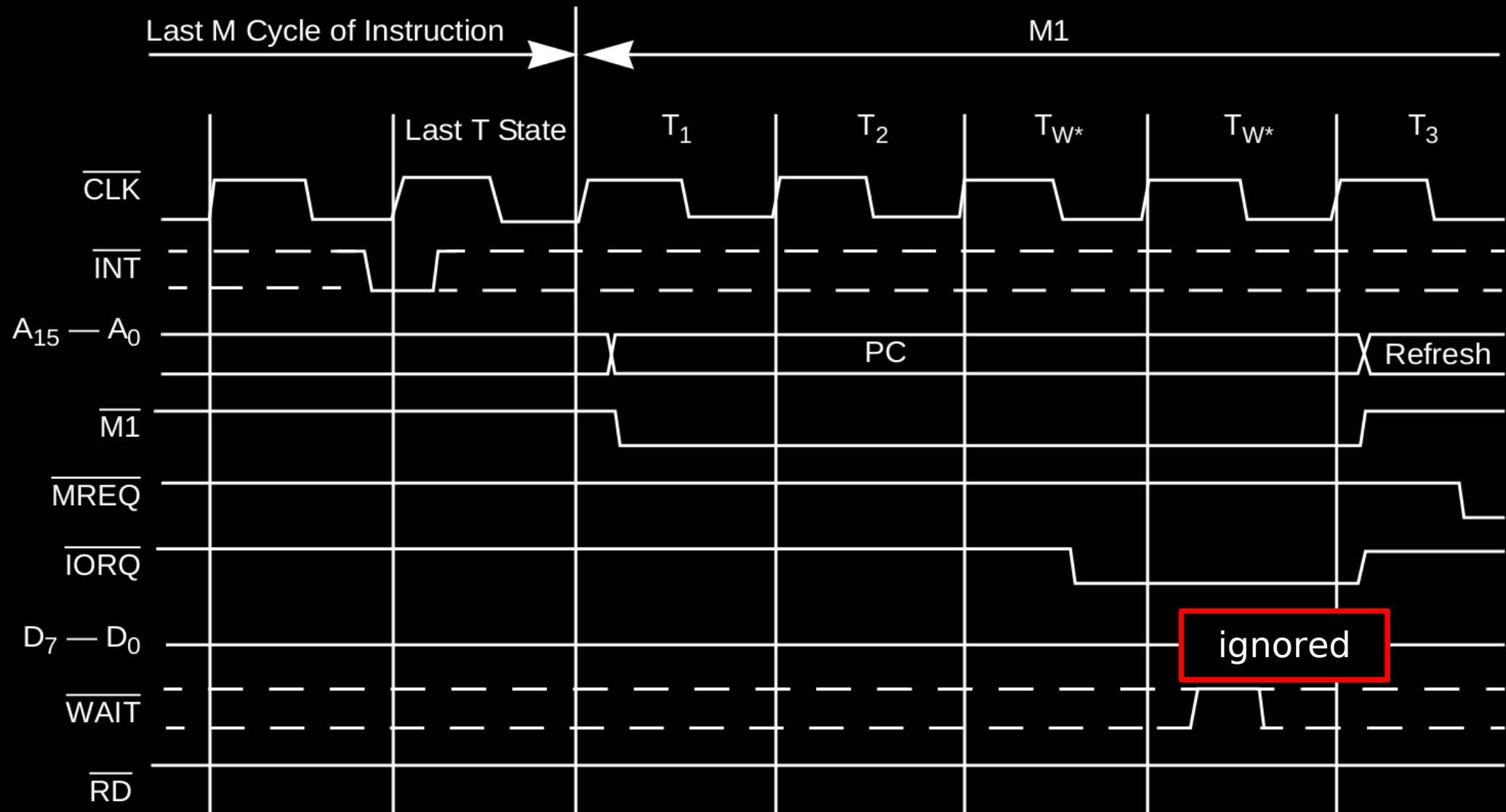






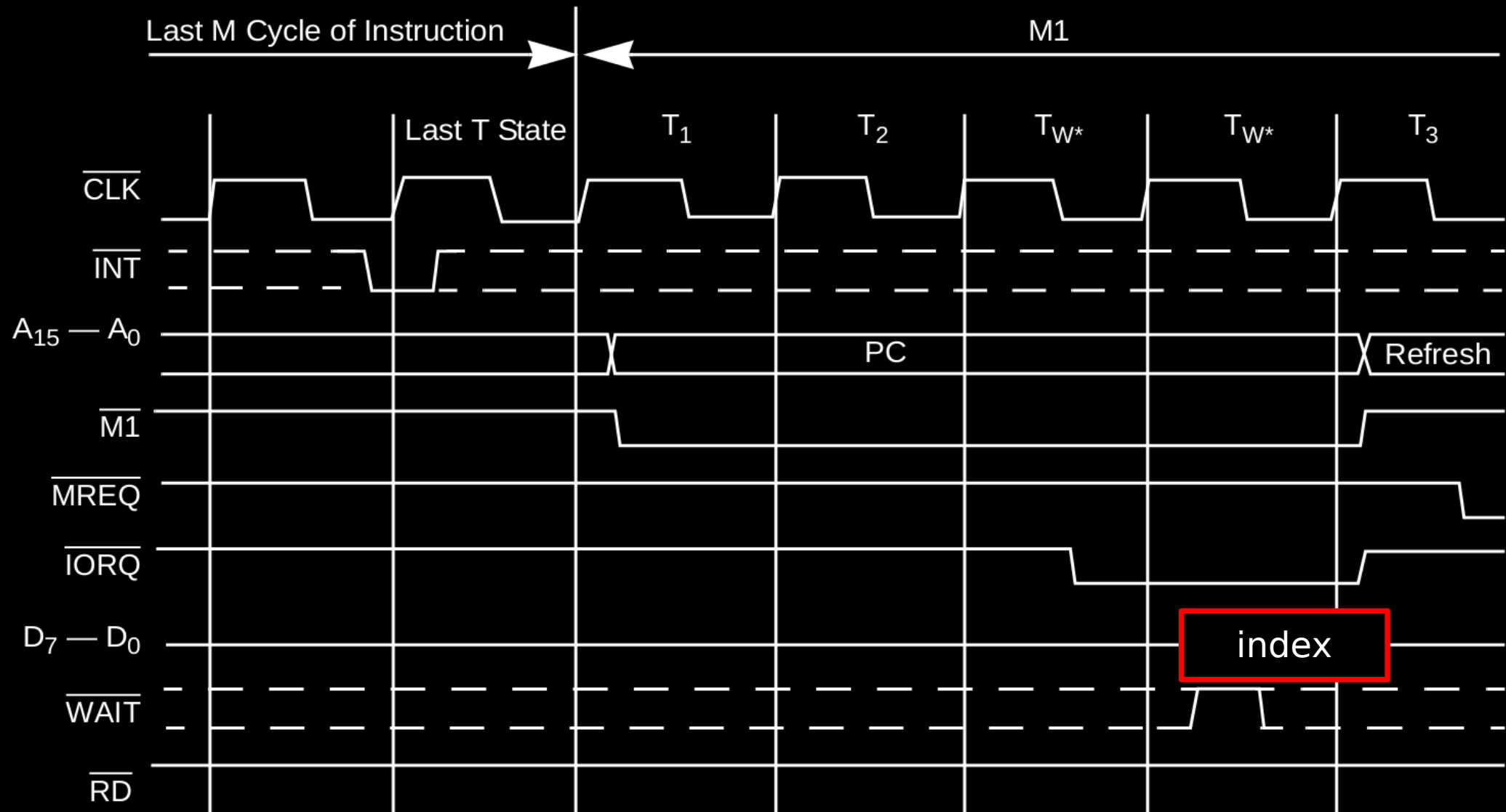
Interrupt mode 0

execute arbitrary instruction



Interrupt mode 1

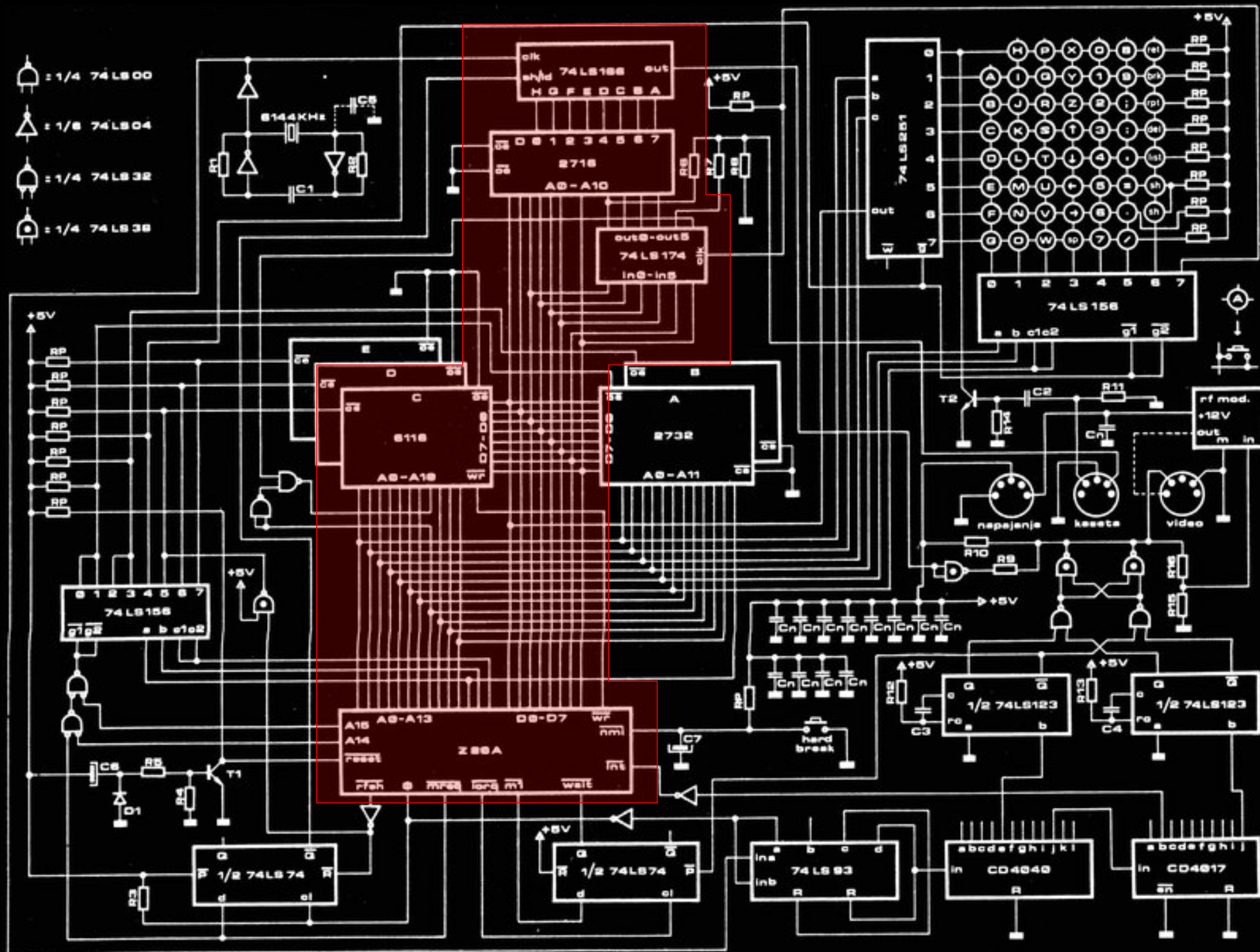
restart to 0x0038



Interrupt mode 2

look-up vector in table pointed by I, jump

-  = 1/4 74LS00
-  = 1/8 74LS04
-  = 1/4 74LS32
-  = 1/4 74LS38



UTV 6007

```
↔ READY  
> BYTE &2BA8, 12  
↔ READY  
> _
```

UNITED®

VIDEO AUDIO

AV1 · AV2

AV · TV

L · H · U

L H U

POWER

000000 sync and overscan area

```
➡ READY  
> BYTE &2BA8, 12  
➡ READY  
> _
```

image

UNITED

VIDEO AUDIO

AV1 · AV2

AV · TV

L · H · U

L H U

POWER

sync and overscan area

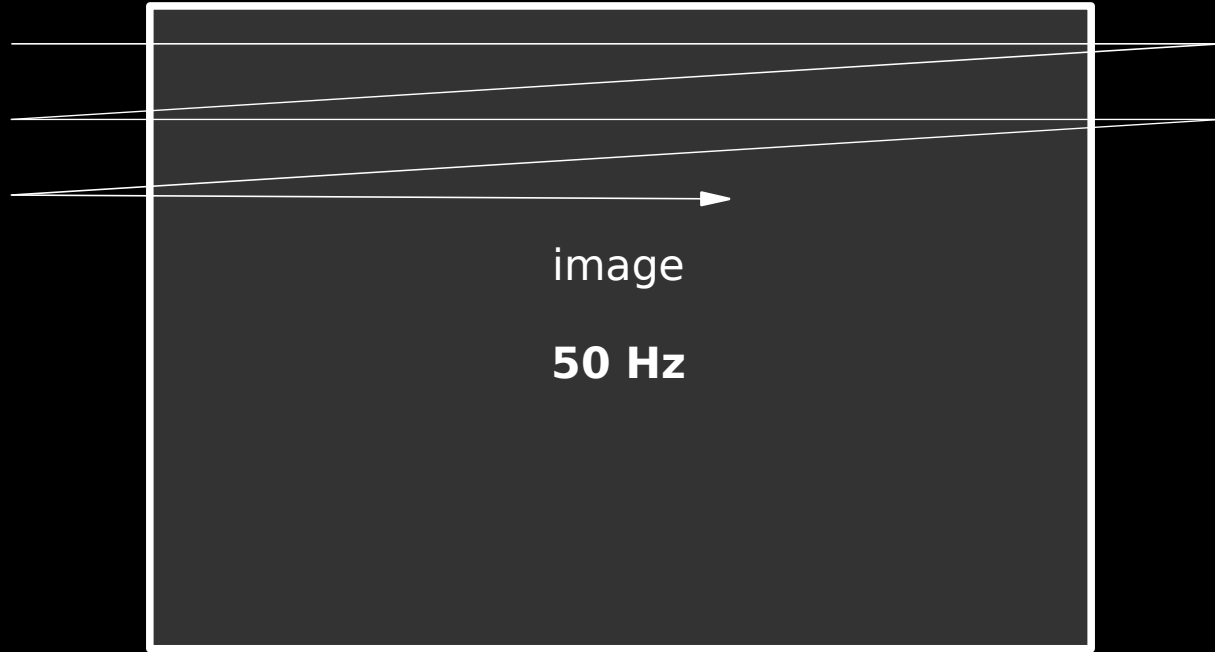
```
> READY  
> BYTE &2BA8.12  
> READY  
> _
```

image

50 Hz

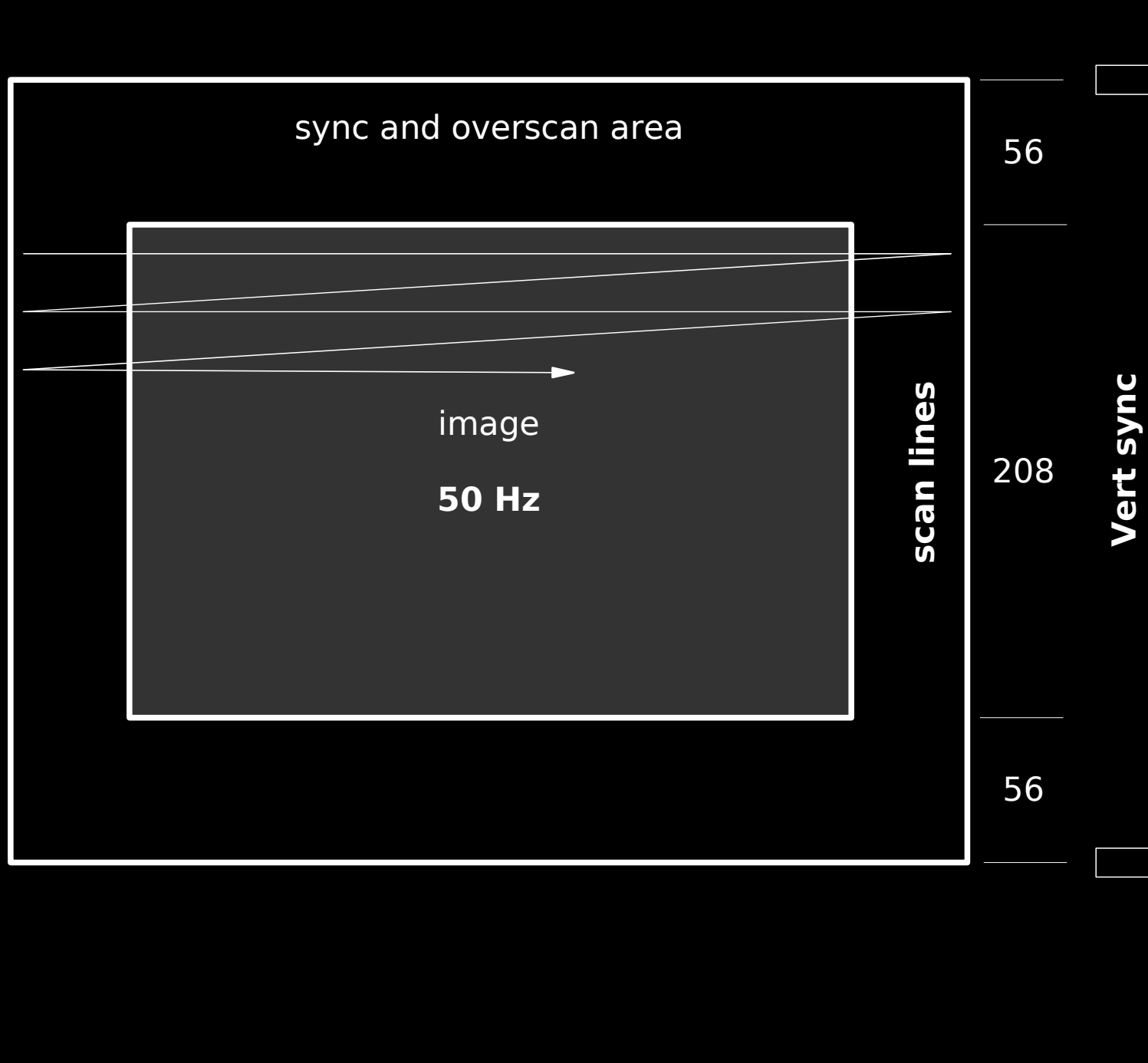
POWER

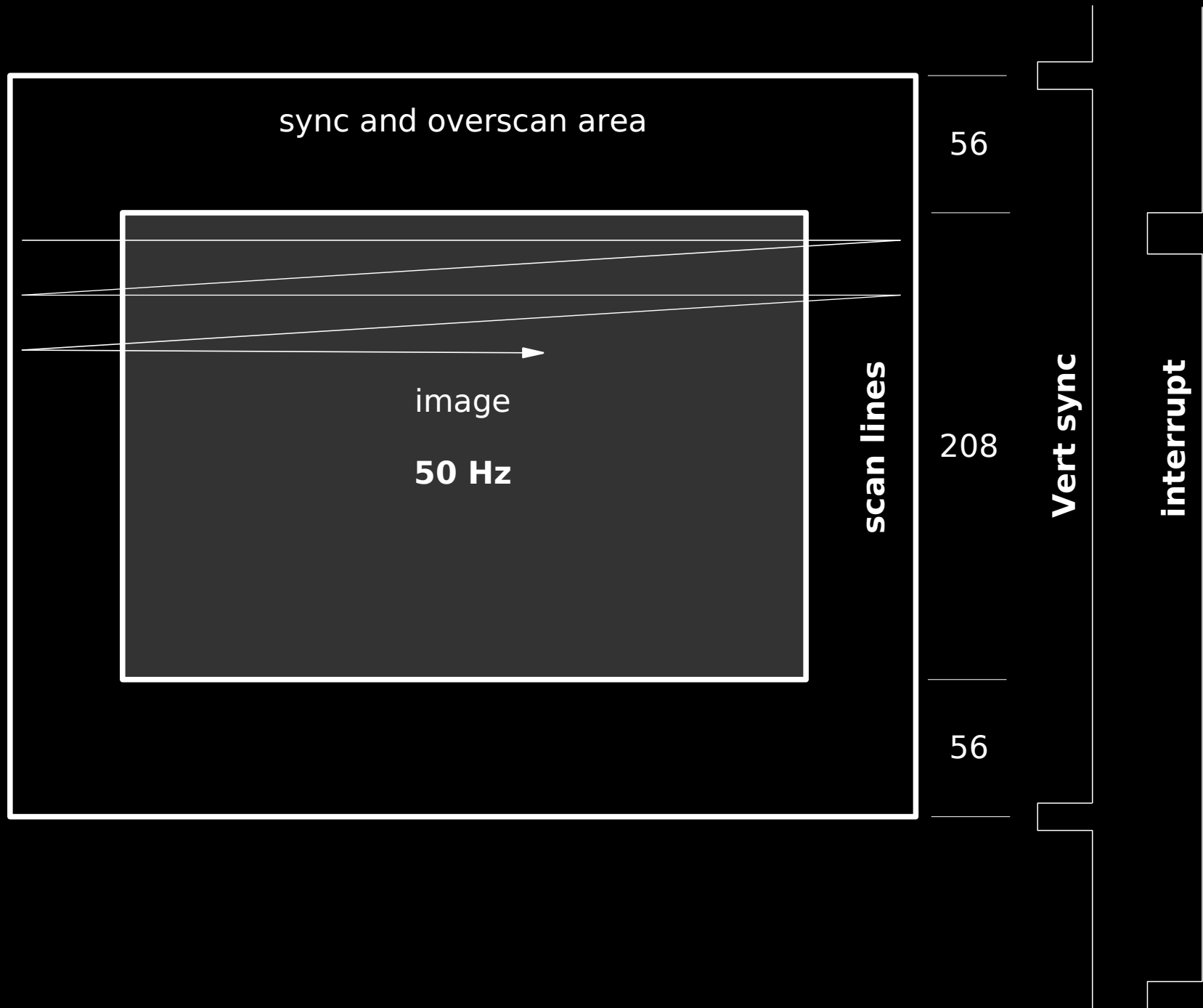
sync and overscan area

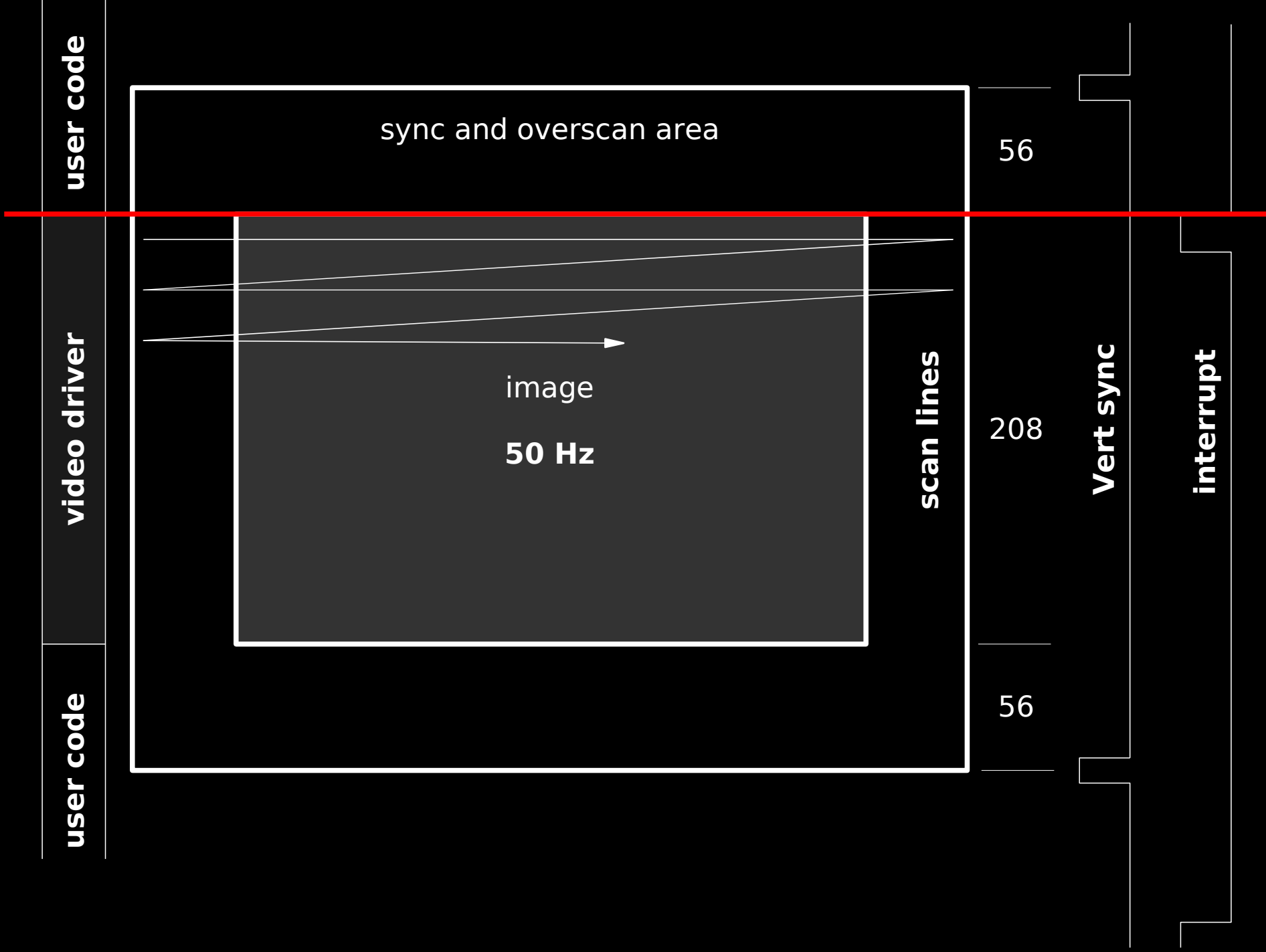


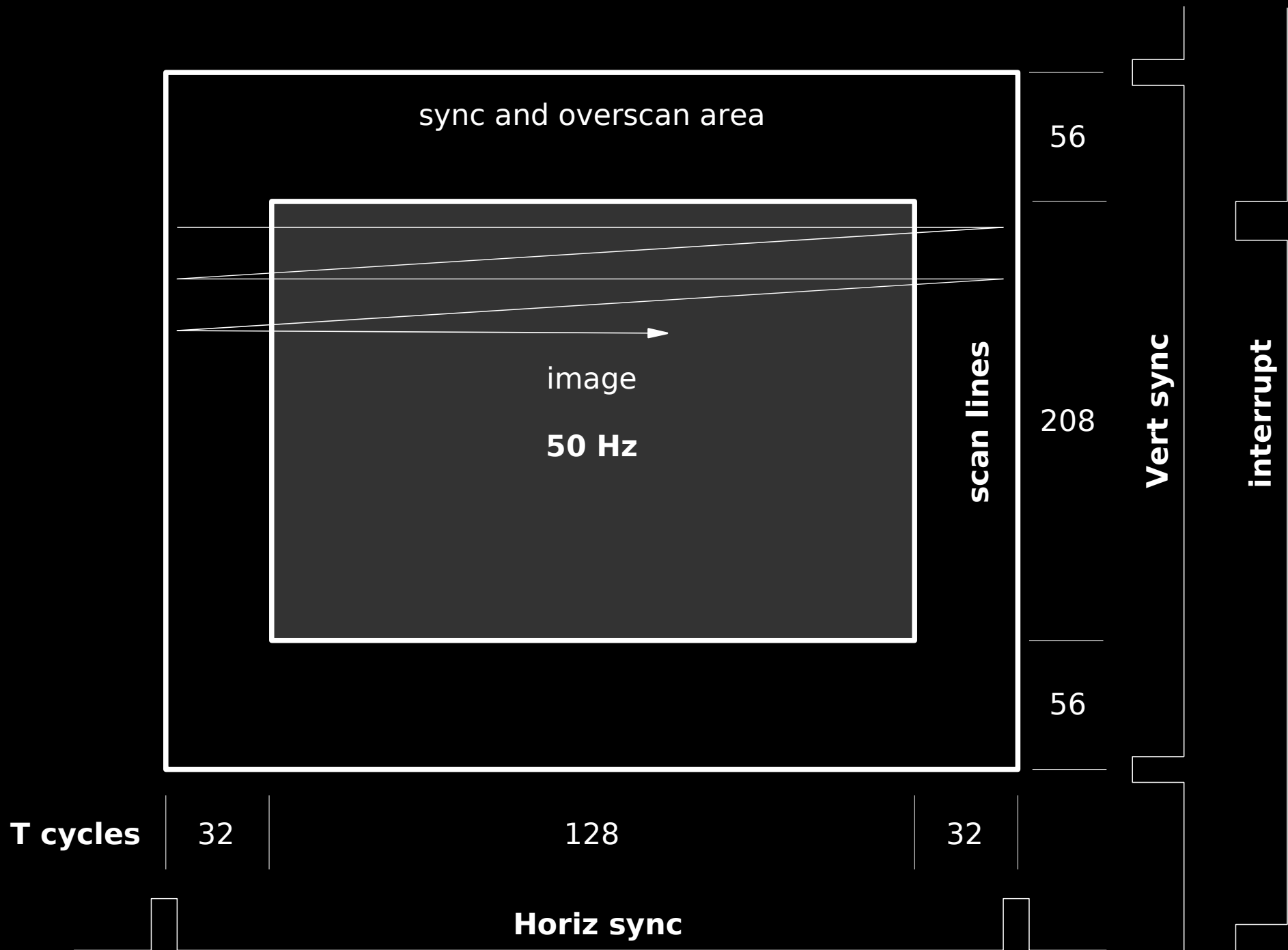
image

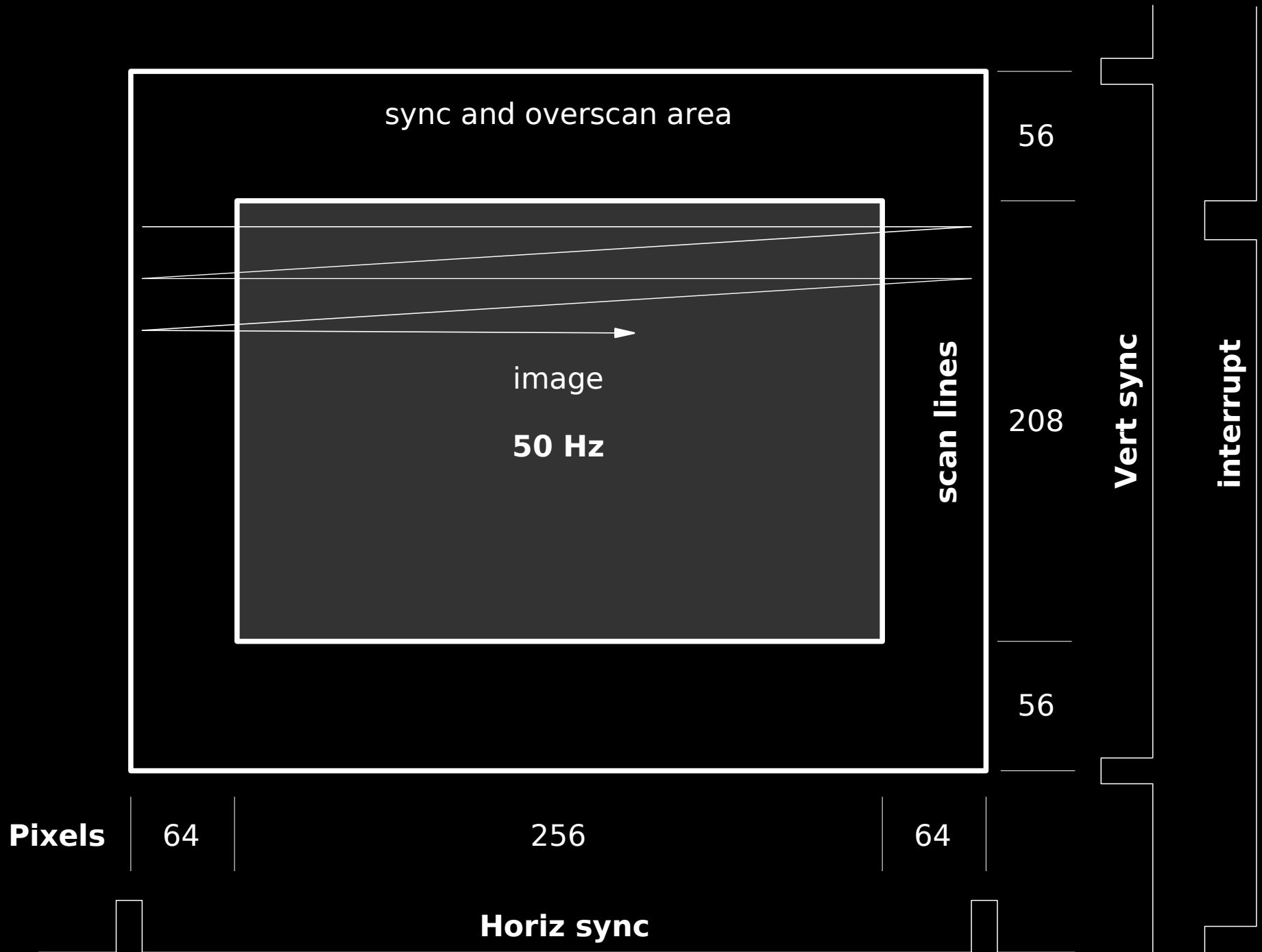
50 Hz







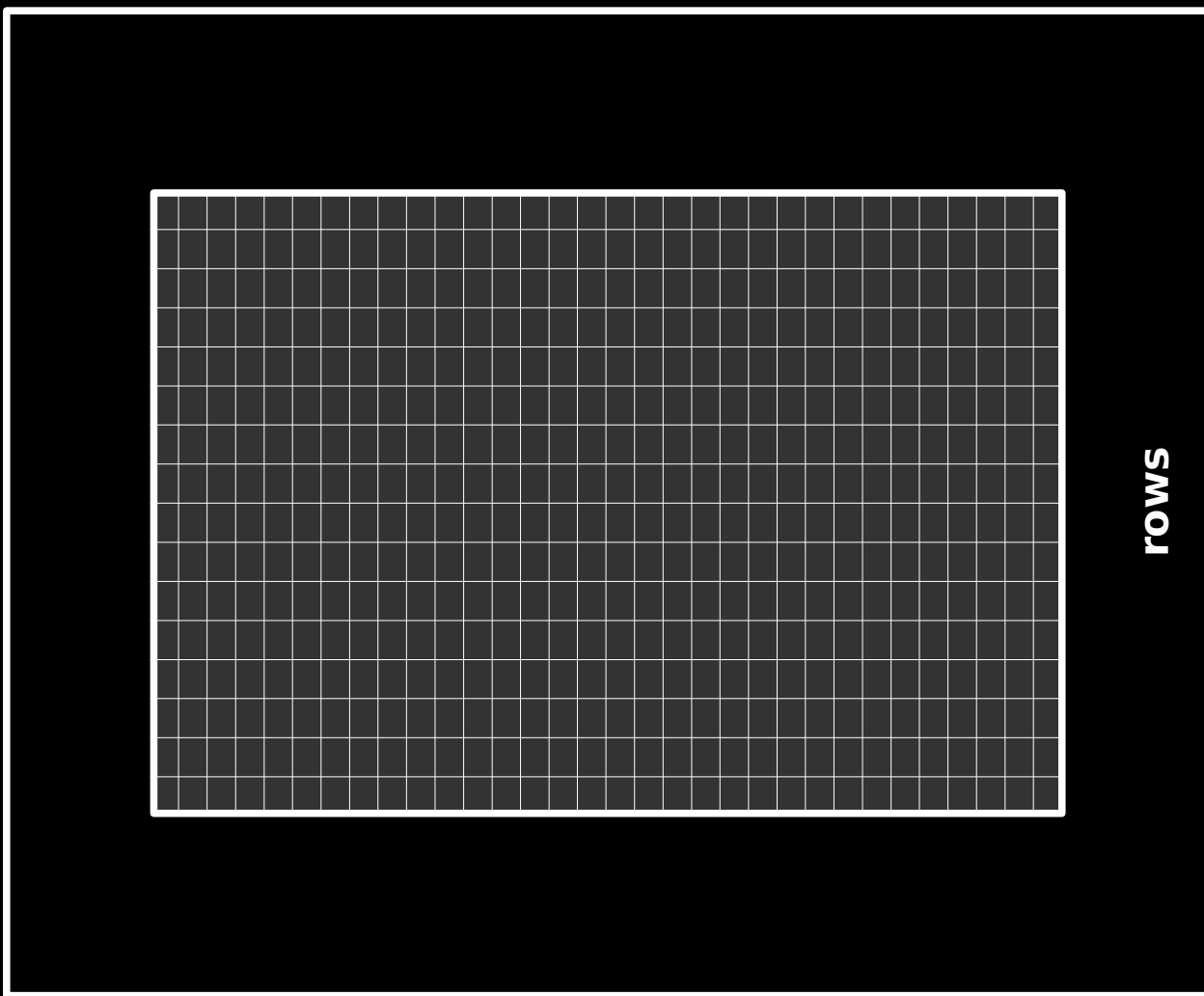








16

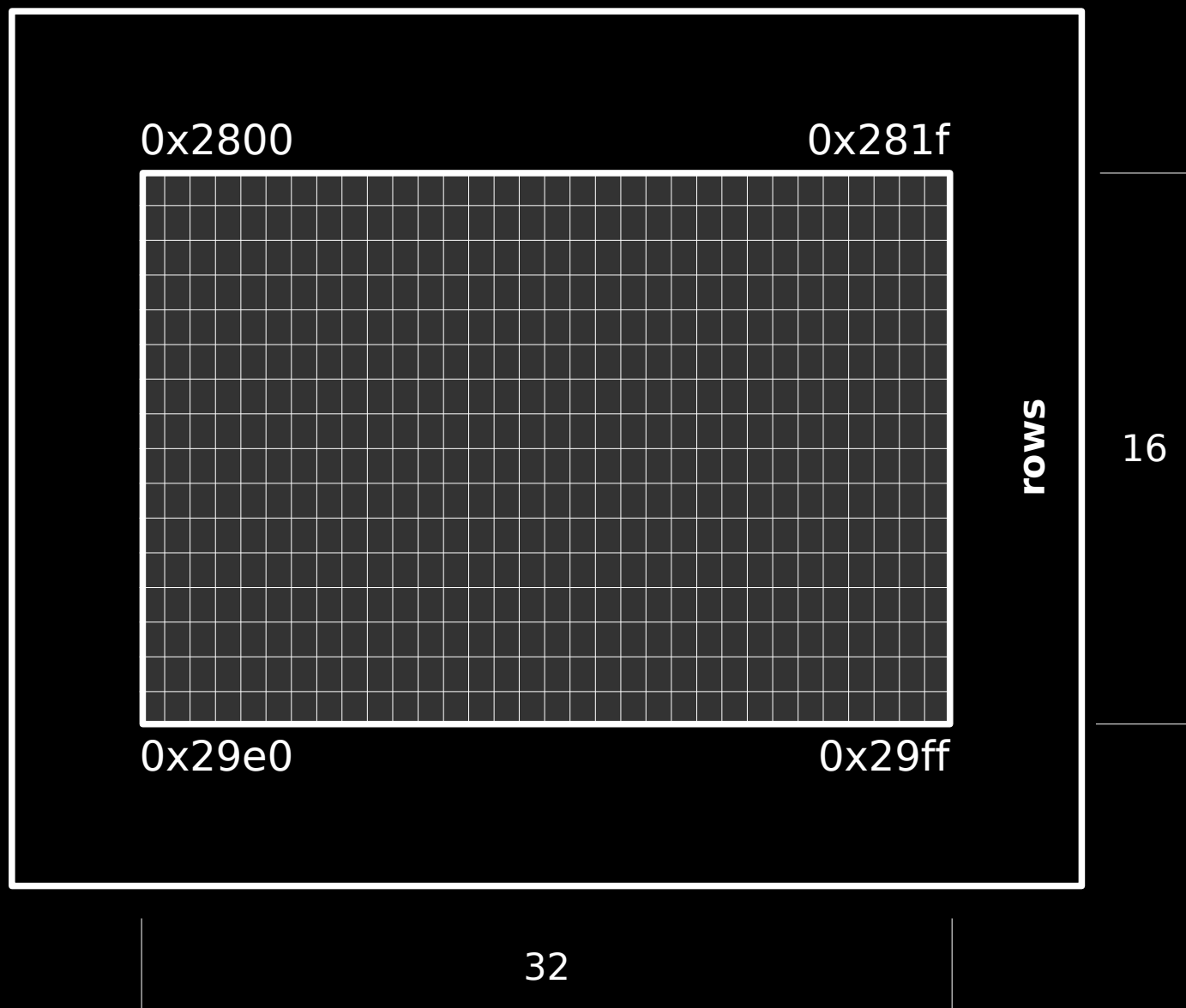


characters

32

rows

16



Character ROM Address

A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	0	1
0	0	1	1	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	1
0	1	0	1	0	0	0	0	0	0	1
0	1	1	0	0	0	0	0	0	0	1
0	1	1	1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	0	0	1
1	0	1	1	0	0	0	0	0	0	1
1	1	0	0	0	0	0	0	0	0	1
1	1	0	1	0	0	0	0	0	0	1
1	1	1	0	0	0	0	0	0	0	1
1	1	1	1	0	0	0	0	0	0	1

Scan line

Character

Data

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	0	0	1	1
1	1	1	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	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

Character ROM Address

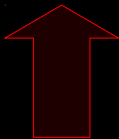
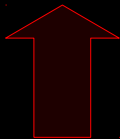
A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	0	1
0	0	1	1	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	1
0	1	0	1	0	0	0	0	0	0	1
0	1	1	0	0	0	0	0	0	0	1
0	1	1	1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	0	0	1
1	0	1	1	0	0	0	0	0	0	1
1	1	0	0	0	0	0	0	0	0	1
1	1	0	1	0	0	0	0	0	0	1
1	1	1	0	0	0	0	0	0	0	1
1	1	1	1	0	0	0	0	0	0	1

Data

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	0	0	1	1
1	1	1	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	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

Scan line

Character



Latch

CPU bus

R3	R2	R1	R0	D7	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----	----	----	----

Character ROM Address

A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	0	1
0	0	1	1	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	1
0	1	0	1	0	0	0	0	0	0	1
0	1	1	0	0	0	0	0	0	0	1
0	1	1	1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	0	0	1
1	0	1	1	0	0	0	0	0	0	1
1	1	0	0	0	0	0	0	0	0	1
1	1	0	1	0	0	0	0	0	0	1
1	1	1	0	0	0	0	0	0	0	1
1	1	1	1	0	0	0	0	0	0	1

Scan line

Character

Data

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	0	0	1	1
1	1	1	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	0	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

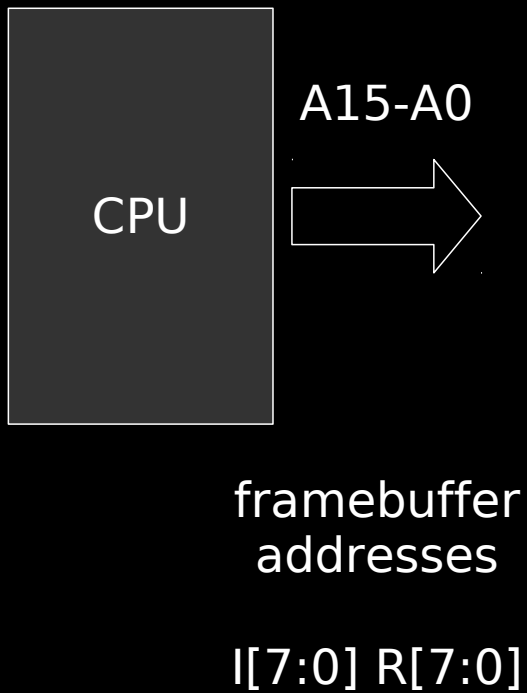
Latch

CPU bus

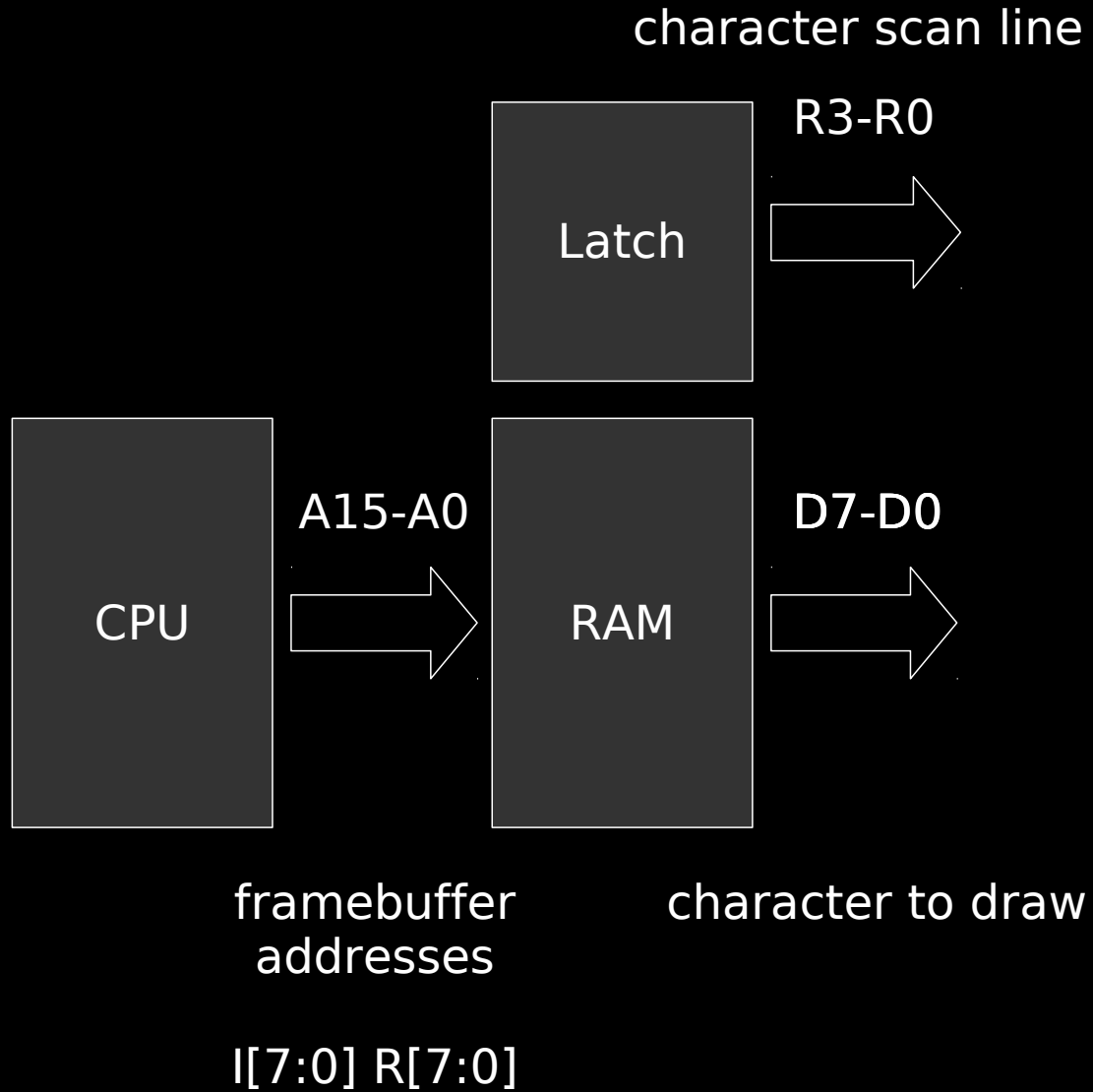
Shift register

R3 R2 R1 R0 D7 D5 D4 D3 D2 D1 D0

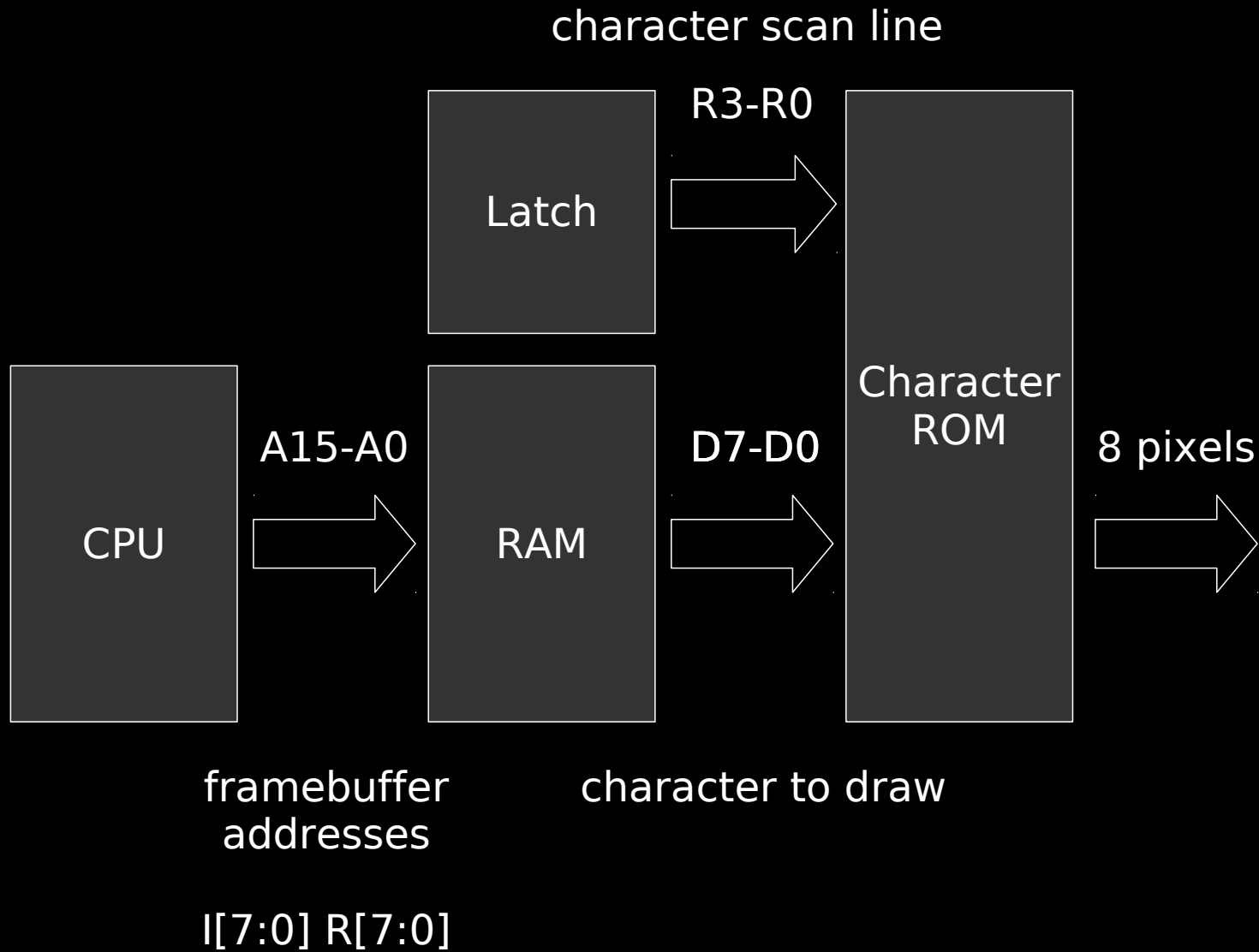
Video pipeline



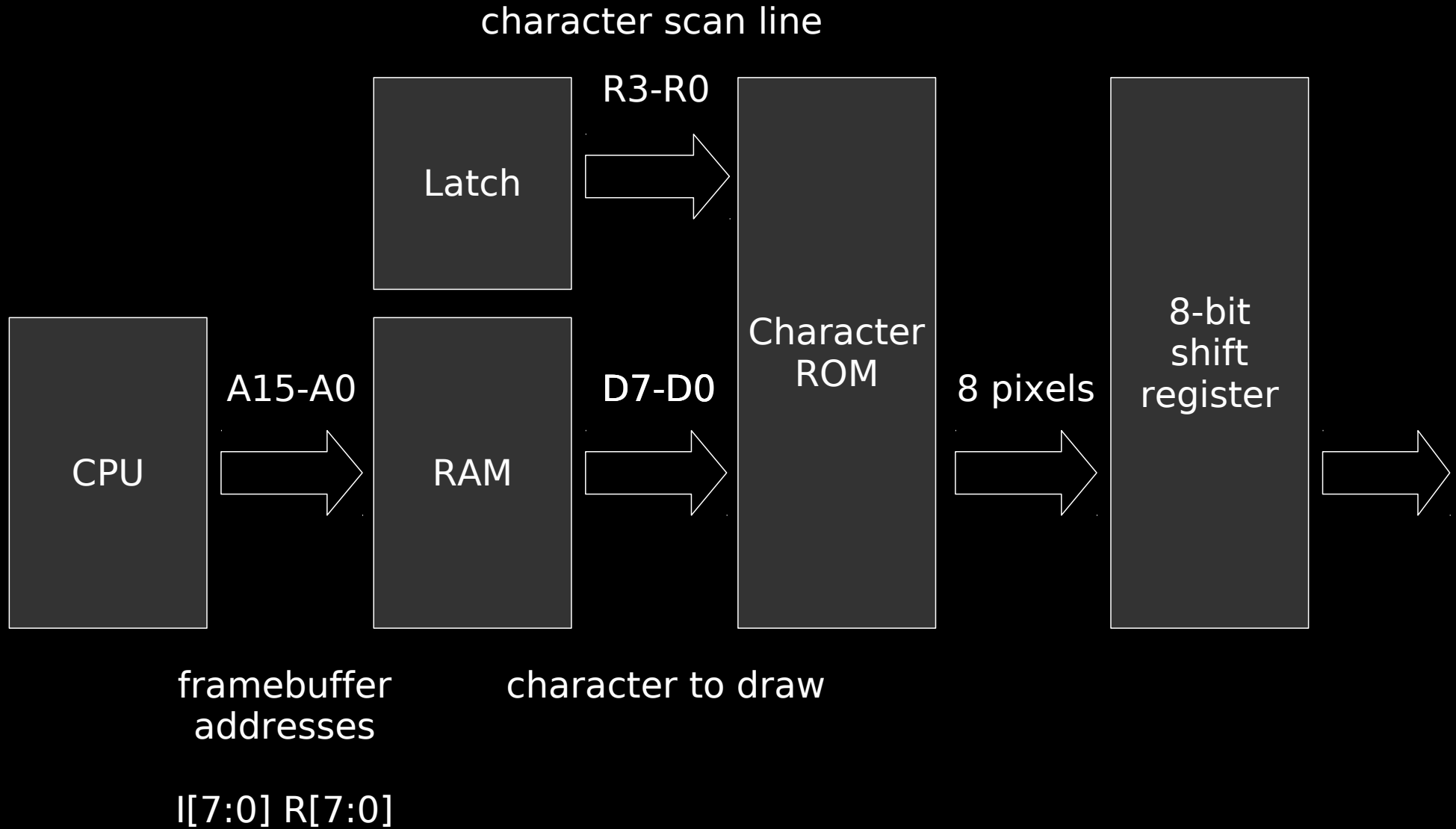
Video pipeline



Video pipeline

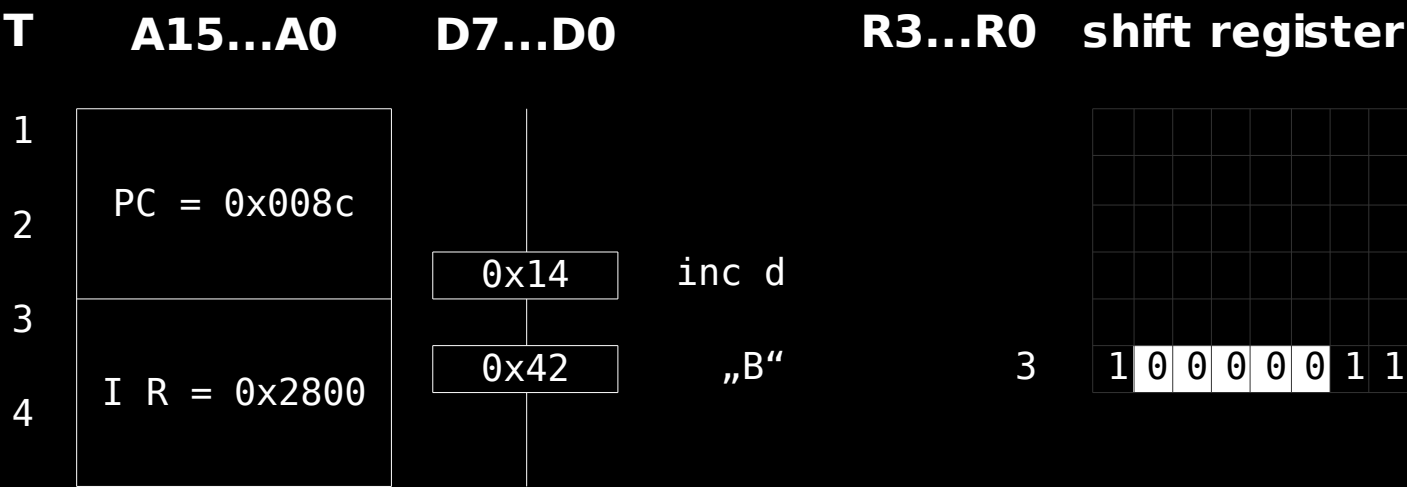


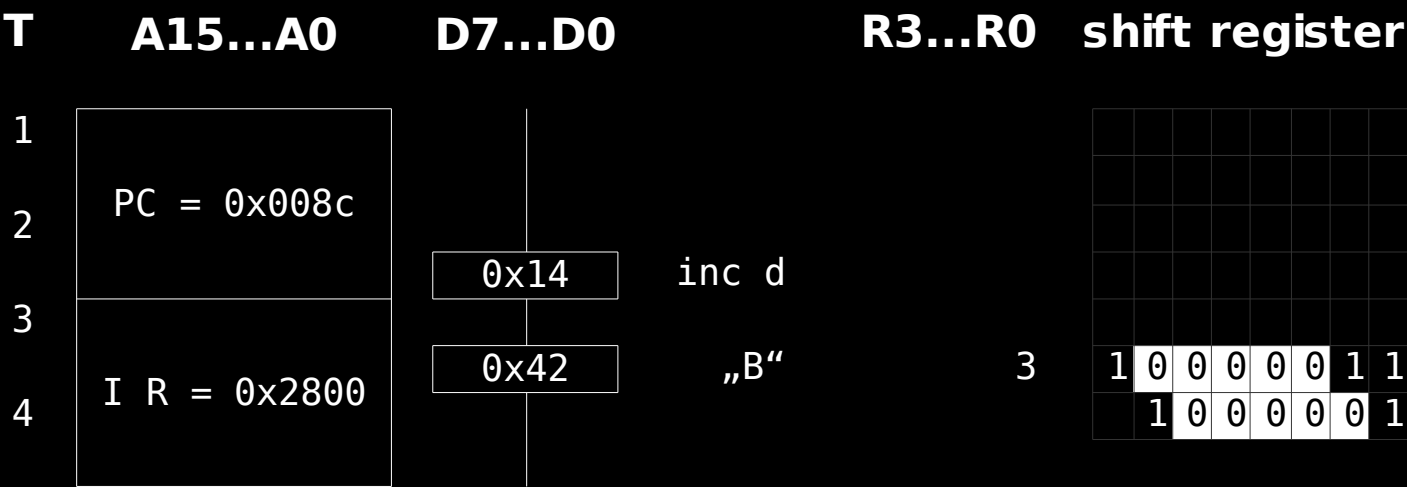
Video pipeline



T A15...A0 D7...D0 R3...R0 shift register







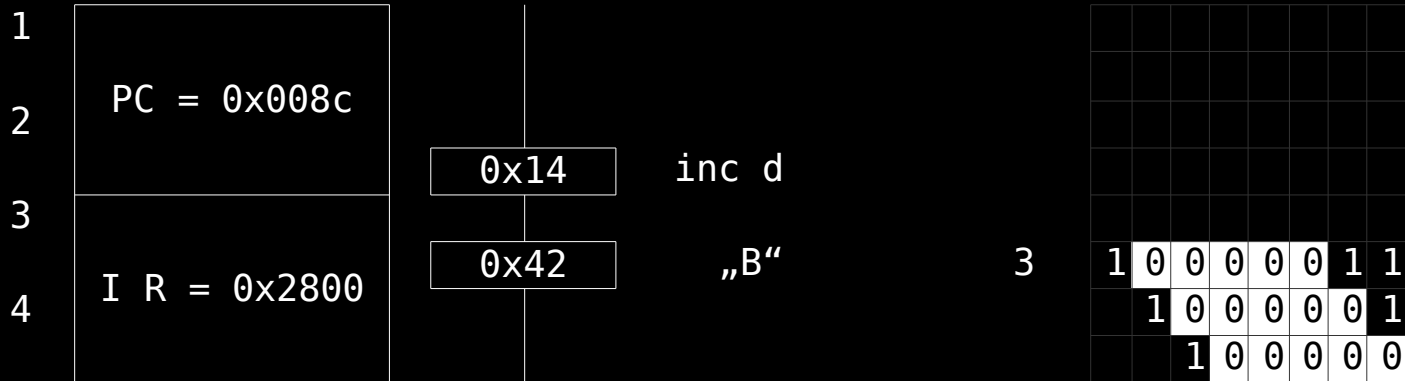
T

A15...A0

D7...D0

R3...R0

shift register



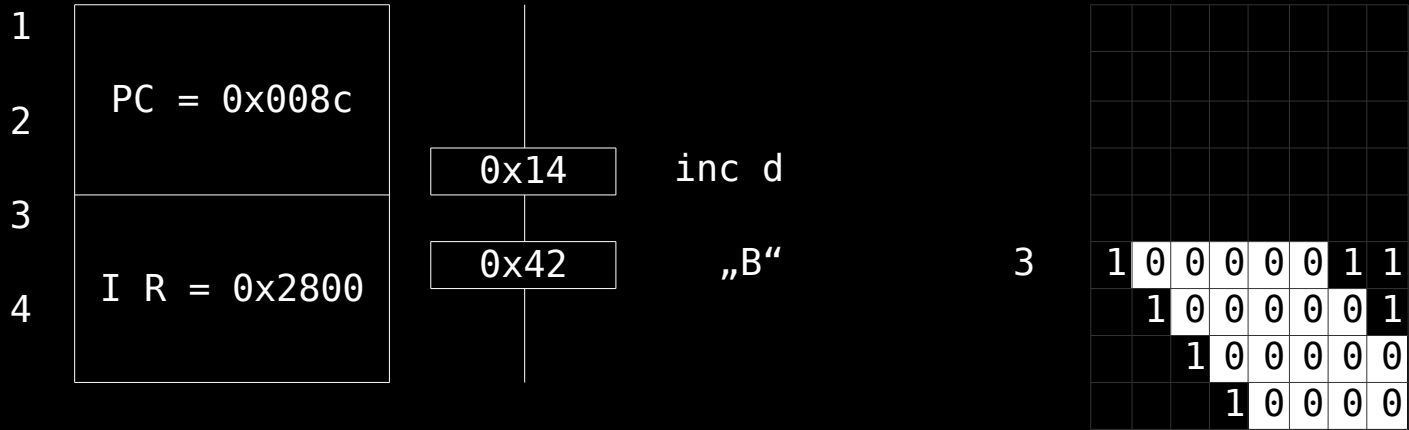
T

A15...A0

D7...D0

R3...R0

shift register



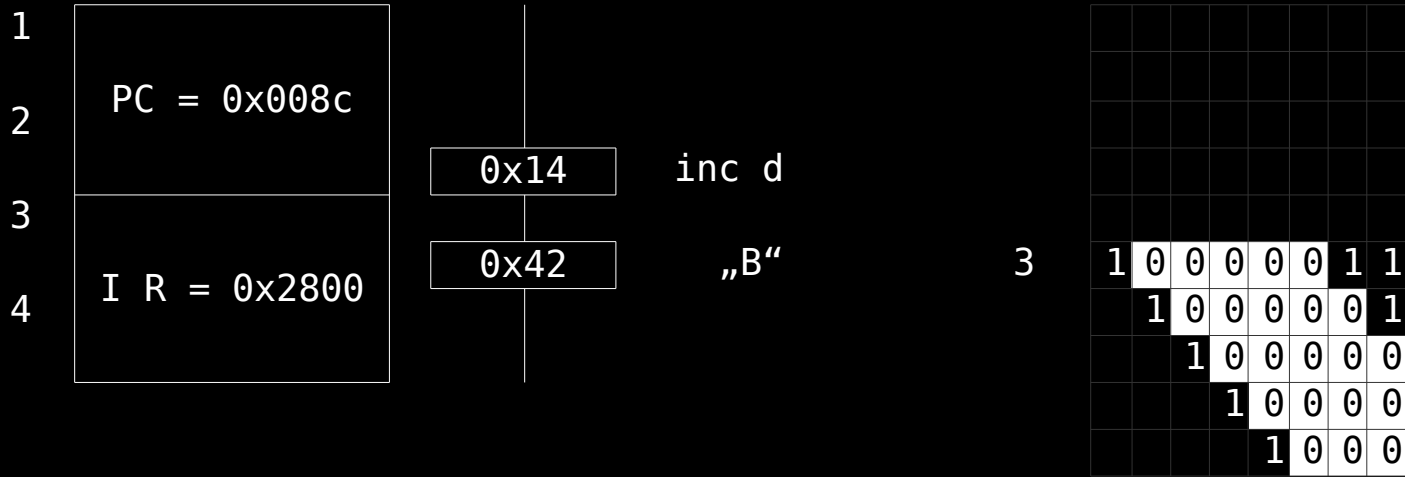
T

A15...A0

D7...D0

R3...R0

shift register



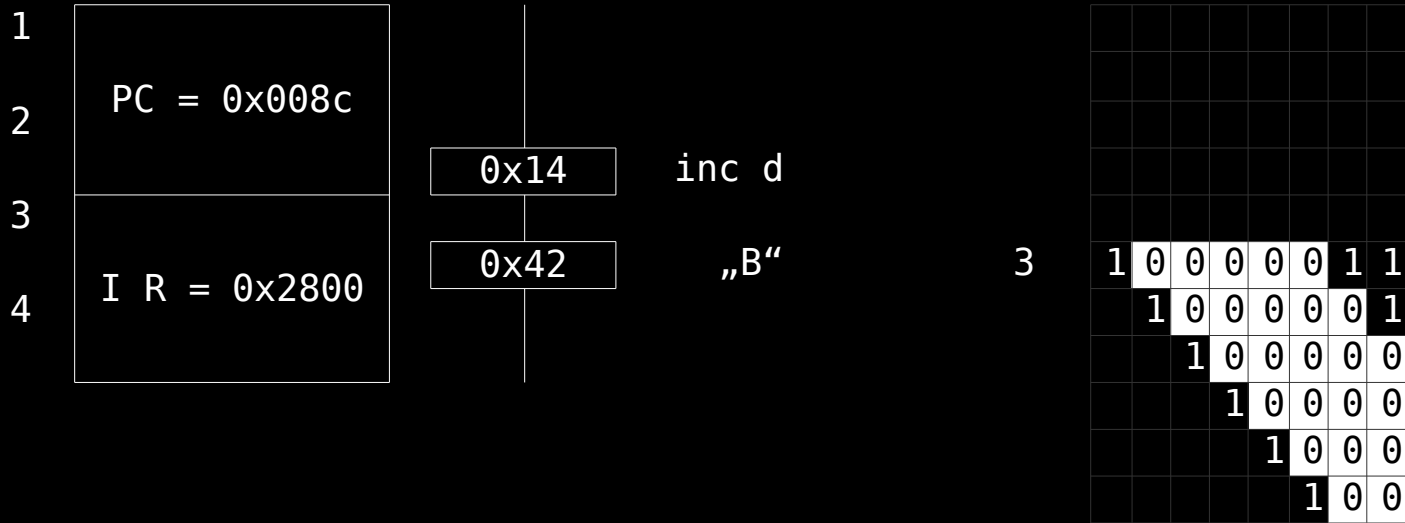
T

A15...A0

D7...D0

R3...R0

shift register



T A15...A0 D7...D0 R3...R0 shift register

1

PC = 0x008c

2

0x14

```
inc d
```

3

I R = 0x2800

0x42

„B“

4

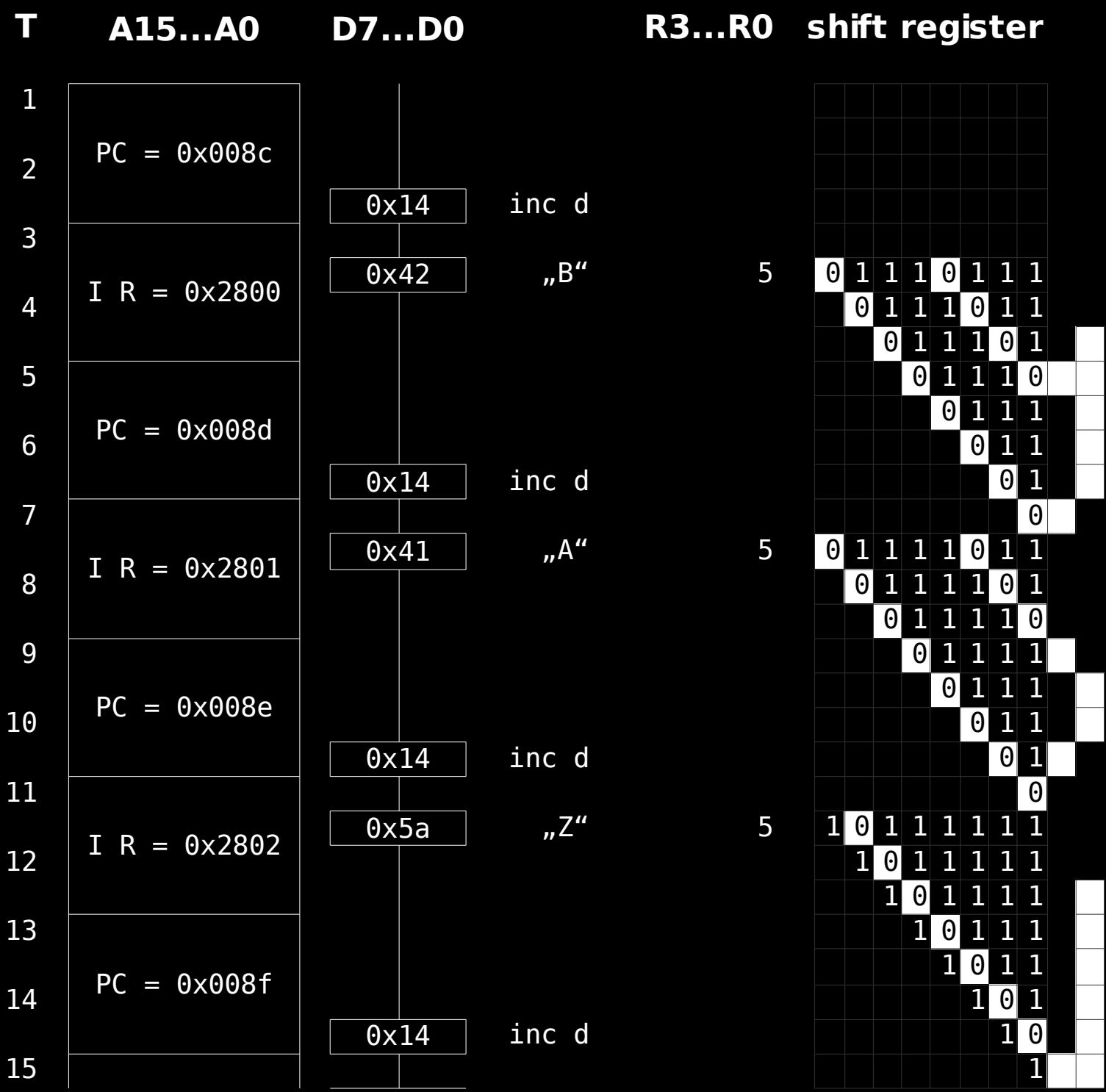
3

1	0	0	0	0	0	1	1
	1	0	0	0	0	0	1
		1	0	0	0	0	0
			1	0	0	0	0
				1	0	0	0
					1	0	0
						1	0
							1

T	A15...A0	D7...D0	R3...R0	shift register
1	PC = 0x008c	0x14	inc d	
2				
3	I R = 0x2800	0x42	„B“	3
4				
5	PC = 0x008d	0x14	inc d	
6				
7	I R = 0x2801	0x41	„A“	3
8				

T	A15...A0	D7...D0	R3...R0	shift register
1	PC = 0x008c	0x14	3	
2				
3				
4	I R = 0x2800	0x42	3	1 0 0 0 0 0 1 1
5				1 0 0 0 0 0 1
6				1 0 0 0 0 0 0
7	PC = 0x008d	0x14	3	1 0 0 0 0 0
8				1 0 0 0 0 0
9				1 0 0 0 0 0
10	PC = 0x008e	0x14	3	1 1 0 0 1 1 1 1
11				1 1 0 0 1 1 1
12				1 1 0 0 1 1
13	I R = 0x2802	0x5a	3	1 1 0 0 1
14				1 1 0 0 0
15				1 1 0 0 0
16	PC = 0x008f	0x14	3	0 0 0 0 0 0 1 1
17				0 0 0 0 0 0 1
18				0 0 0 0 0 0 0
19				0 0 0 0 0 0 0
20				0 0 0 0 0 0 0
21				0 0 0 0 0 0 0
22				0 0 0 0 0 0 0
23				0 0 0 0 0 0 0
24				0 0 0 0 0 0 0
25				0 0 0 0 0 0 0
26				0 0 0 0 0 0 0
27				0 0 0 0 0 0 0

T	A15...A0	D7...D0	R3...R0	shift register
1	PC = 0x008c	0x14		
2				
3	I R = 0x2800	0x42	4	01110111
4				
5	PC = 0x008d	0x14		
6				
7	I R = 0x2801	0x41	4	10110111
8				
9	PC = 0x008e	0x14		
10				
11	I R = 0x2802	0x5a	4	01111111
12				
13	PC = 0x008f	0x14		
14				
15				



T	A15...A0	D7...D0	R3...R0	shift register
1	PC = 0x008c	0x14		
2				
3	I R = 0x2800	0x42	6	01110111
4				
5	PC = 0x008d	0x14		
6				
7	I R = 0x2801	0x41	6	01111011
8				
9	PC = 0x008e	0x14		
10				
11	I R = 0x2802	0x5a	6	11011111
12				
13	PC = 0x008f	0x14		
14				
15				

T	A15...A0	D7...D0	R3...R0	shift register
1	PC = 0x008c	0x14	inc d	
2				
3				
4	I R = 0x2800	0x42	„B“	7
5				1 0 0 0 0 1 1 1
6				1 0 0 0 0 1 1
7	PC = 0x008d	0x14	inc d	1 0 0 0 0 1
8				1 0 0 0 0
9				1 0 0 0
10	PC = 0x008e	0x14	inc d	1 0
11				1
12				0 1 1 1 1 0 1 1
13	I R = 0x2802	0x5a	„Z“	7
14				0 1 1 1 1 0 1
15				0 1 1 1 1 0
16	PC = 0x008f	0x14	inc d	0 1 1 1 1
17				0 1 1 1
18				0 1 1
19				1 1 1 0 1 1 1 1
20				1 1 1 0 1 1 1
21				1 1 1 0 1 1
22				1 1 1 0 1
23				1 1 1 0
24				1 1 1
25				1 1 1
26				1 1
27				1

32		48	0	64	€	80	P
33	!	49	1	65	A	81	Q
34	"	50	2	66	B	82	R
35	#	51	3	67	C	83	S
36	\$	52	4	68	D	84	T
37	%	53	5	69	E	85	U
38	&	54	6	70	F	86	V
39	'	55	7	71	G	87	W
40	(	56	8	72	H	88	X
41	)	57	9	73	I	89	Y
42	*	58	:	74	J	90	Z
43	+	59	;	75	K	91	[
44	,	60	<	76	L	92	\
45	-	61	=	77	M	93	]
46	.	62	>	78	N	94	^
47	/	63	?	79	O	95	_

Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex								
0	00	NUL	16	10	DLE	32	20		48	30	0	64	40	@	80	50	P	96	60	`	112	70	p
1	01	SOH	17	11	DC1	33	21	!	49	31	1	65	41	A	81	51	Q	97	61	a	113	71	q
2	02	STX	18	12	DC2	34	22	"	50	32	2	66	42	B	82	52	R	98	62	b	114	72	r
3	03	ETX	19	13	DC3	35	23	#	51	33	3	67	43	C	83	53	S	99	63	c	115	73	s
4	04	EOT	20	14	DC4	36	24	\$	52	34	4	68	44	D	84	54	T	100	64	d	116	74	t
5	05	ENQ	21	15	NAK	37	25	%	53	35	5	69	45	E	85	55	U	101	65	e	117	75	u
6	06	ACK	22	16	SYN	38	26	&	54	36	6	70	46	F	86	56	V	102	66	f	118	76	v
7	07	BEL	23	17	ETB	39	27	'	55	37	7	71	47	G	87	57	W	103	67	g	119	77	w
8	08	BS	24	18	CAN	40	28	(	56	38	8	72	48	H	88	58	X	104	68	h	120	78	x
9	09	HT	25	19	EM	41	29	)	57	39	9	73	49	I	89	59	Y	105	69	i	121	79	y
10	0A	LF	26	1A	SUB	42	2A	*	58	3A	:	74	4A	J	90	5A	Z	106	6A	j	122	7A	z
11	0B	VT	27	1B	ESC	43	2B	+	59	3B	;	75	4B	K	91	5B	[	107	6B	k	123	7B	{
12	0C	FF	28	1C	FS	44	2C	,	60	3C	<	76	4C	L	92	5C	\	108	6C	l	124	7C	
13	0D	CR	29	1D	GS	45	2D	-	61	3D	=	77	4D	M	93	5D	]	109	6D	m	125	7D	}
14	0E	S0	30	1E	RS	46	2E	.	62	3E	>	78	4E	N	94	5E	^	110	6E	n	126	7E	~
15	0F	SI	31	1F	US	47	2F	/	63	3F	?	79	4F	O	95	5F	_	111	6F	o	127	7F	DEL

128		144		160		176	
129	.	145	.	161	.	177	.
130	.	146	.	162	.	178	.
131	.	147	.	163	.	179	.
132	.	148	.	164	.	180	.
133	.	149	.	165	.	181	.
134	.	150	.	166	.	182	.
135	.	151	.	167	.	183	.
136	.	152	.	168	.	184	.
137	.	153	.	169	.	185	.
138	.	154	.	170	.	186	.
139	.	155	.	171	.	187	.
140	.	156	.	172	.	188	.
141	.	157	.	173	.	189	.
142	.	158	.	174	.	190	.
143	.	159	.	175	.	191	.

pseudo-graphics

32 x 16 characters

2 x 3 dots/character

64 x 48 dots

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

1111111111111111

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

— 100 —

1111111111111111

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36

[illegible]

00130

HI 00357

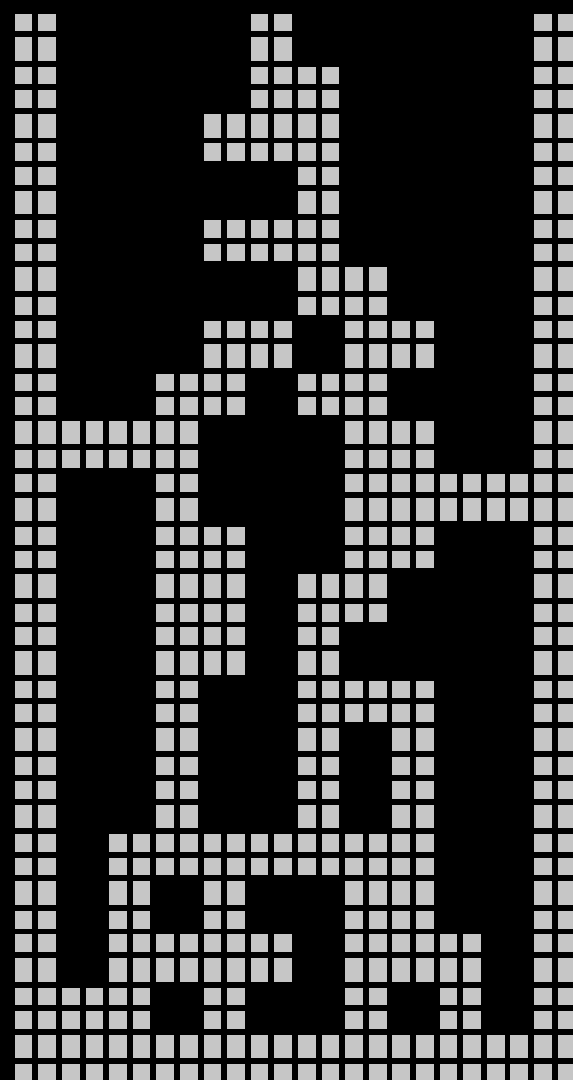
SC 00357

LEVEL 1

NEXT:

PRESS
ENTER

T E T A I S



PROGRAM BY

DRAGOLJUB
OBRADOVIĆ

21.7.1988.

■ = 00000

GAME
OVER



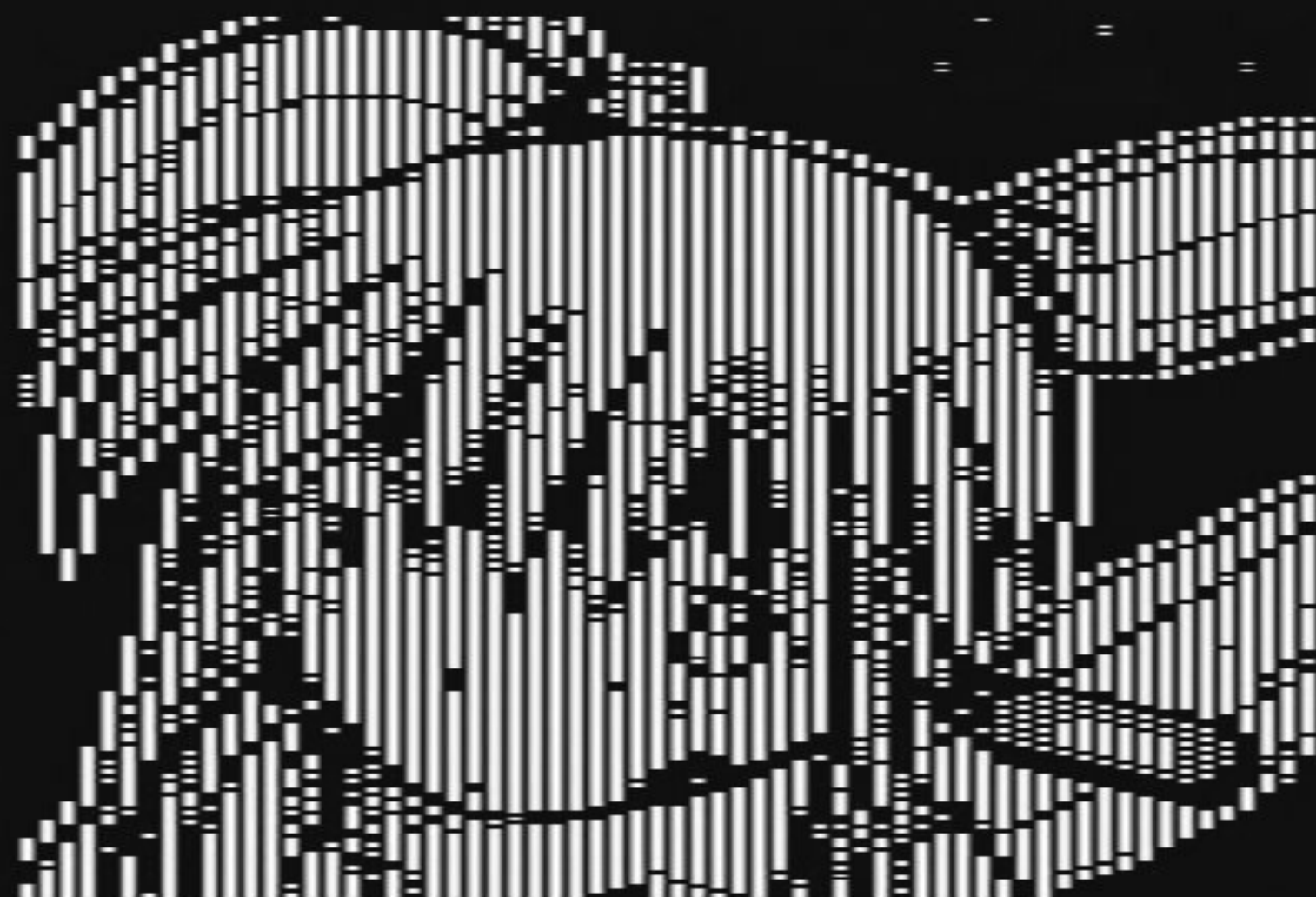
INSPECTOR
SPRINTON

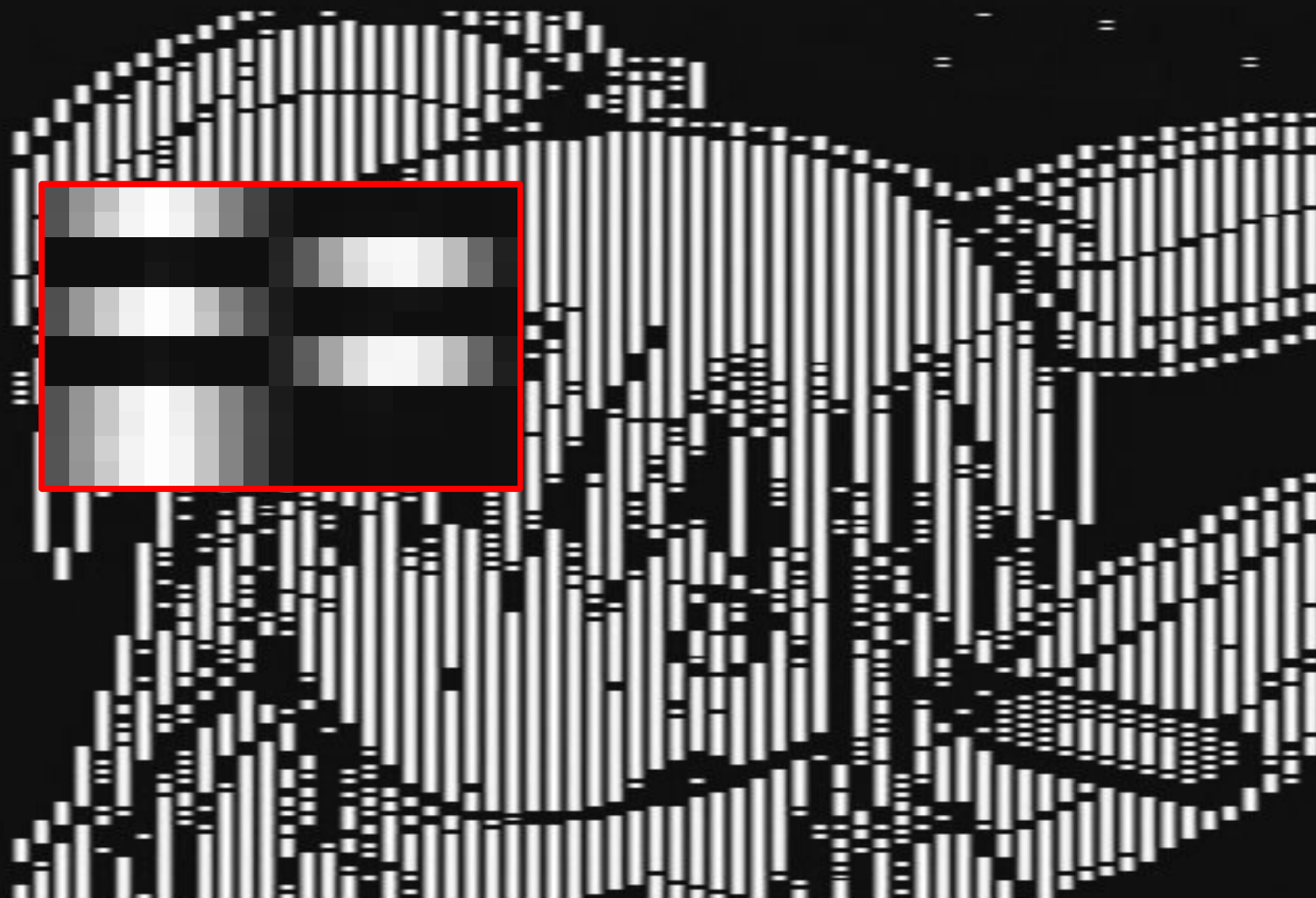
SLUČAJ
PRVI!
BY
FRANKIE
SOFTWARE

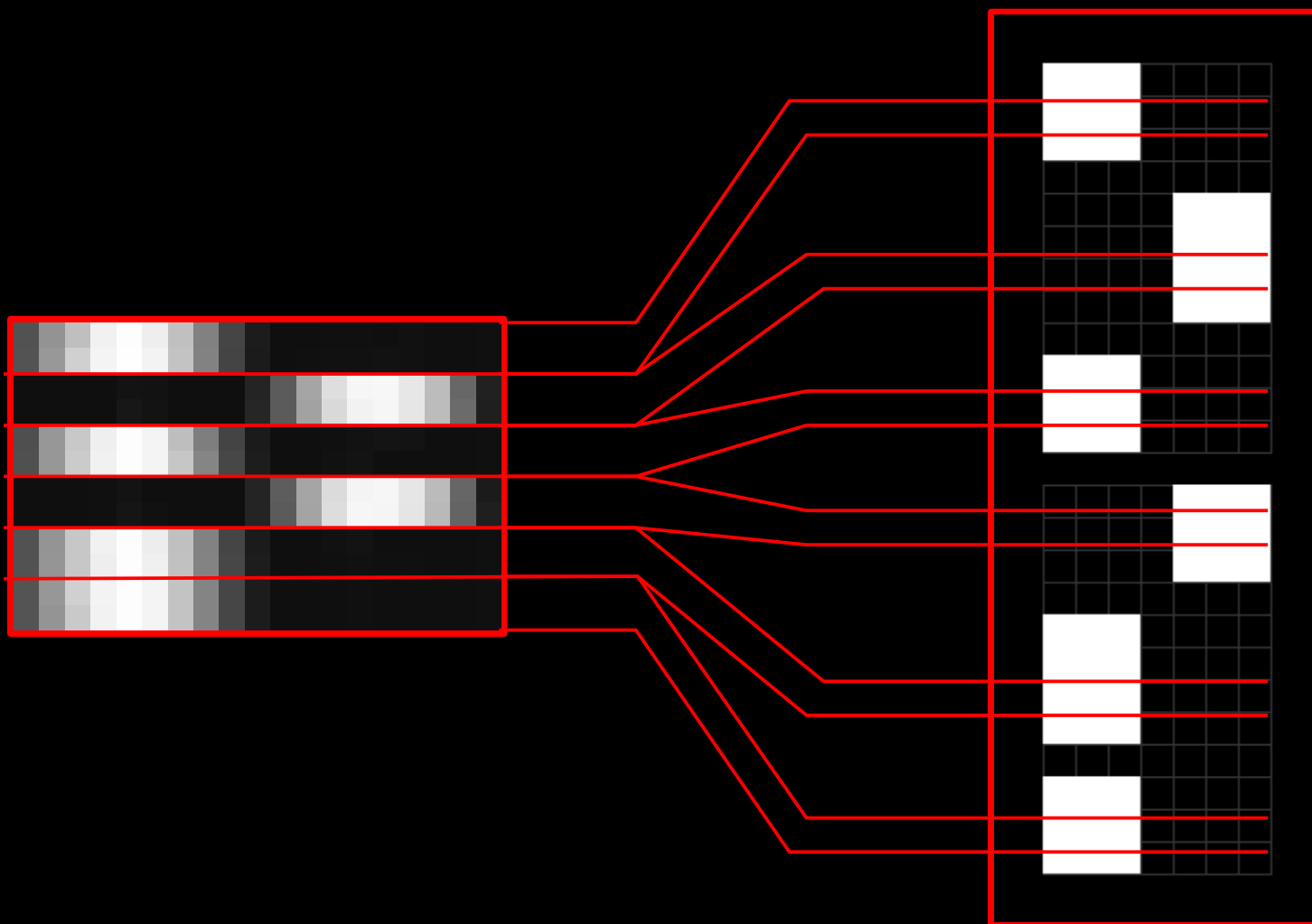
<BLANKO>

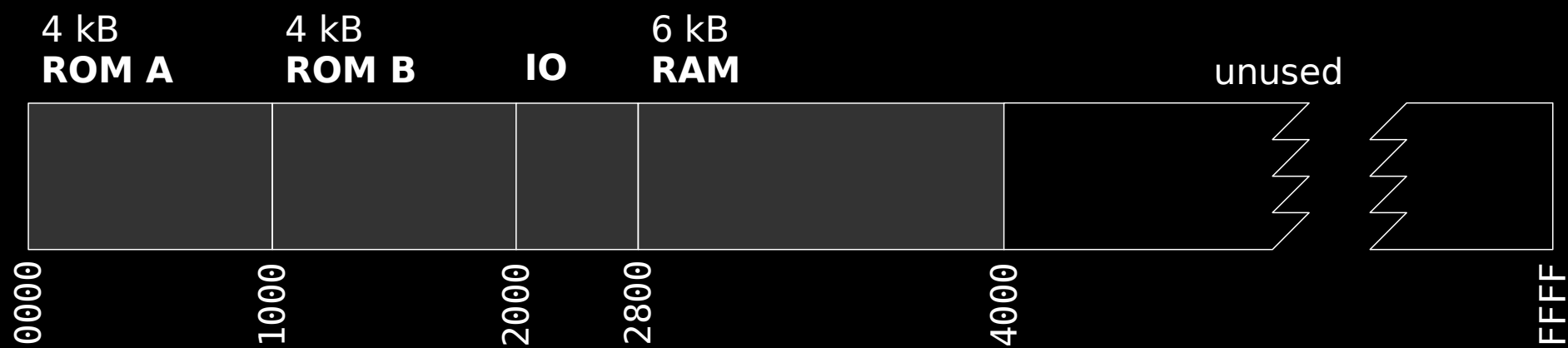


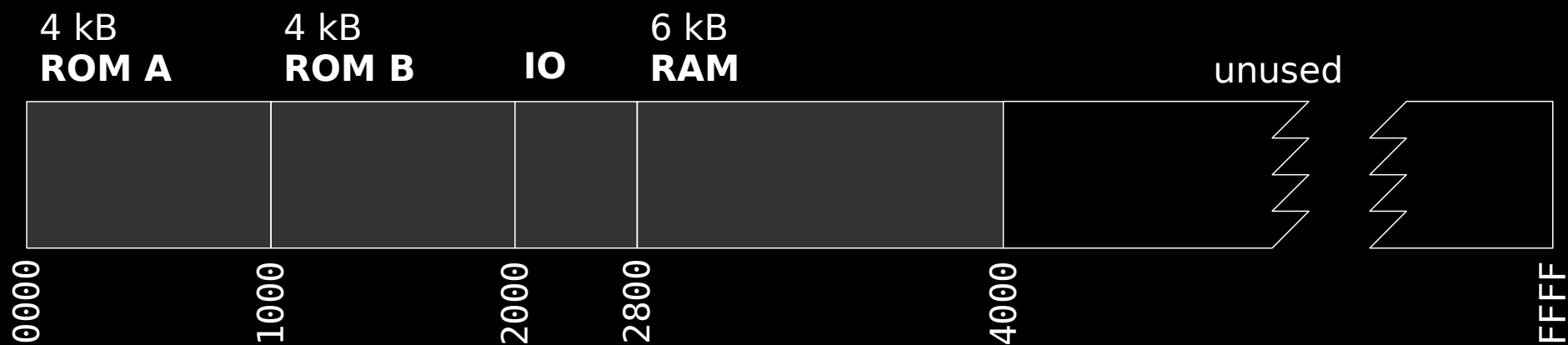
D A N C I N G D E M O N

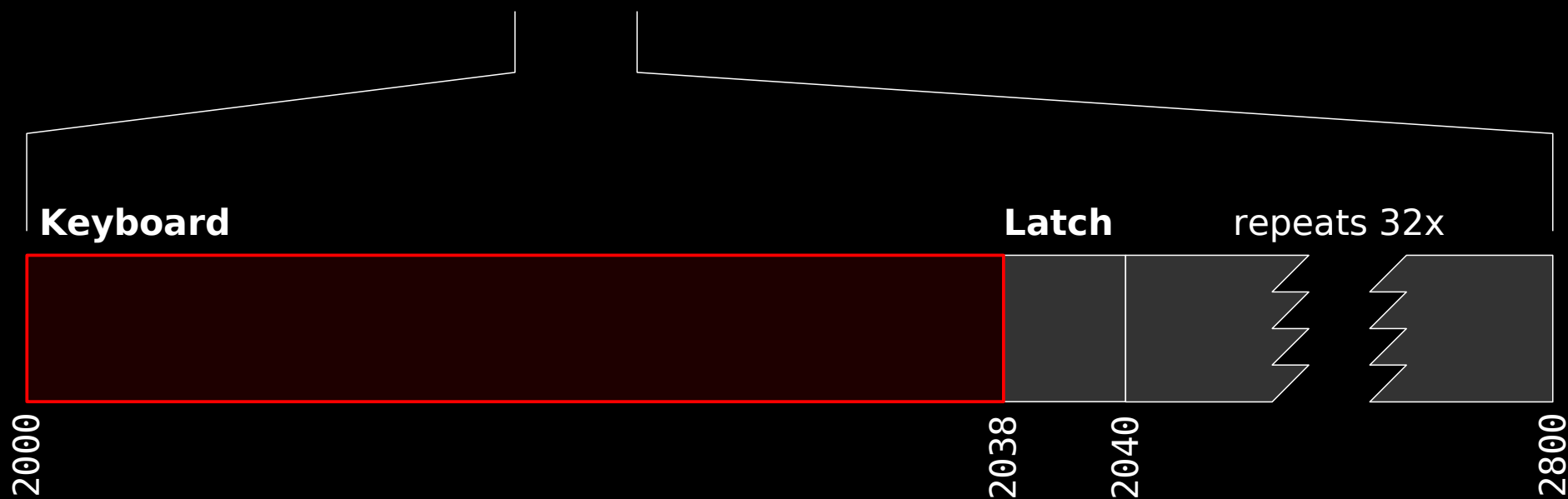
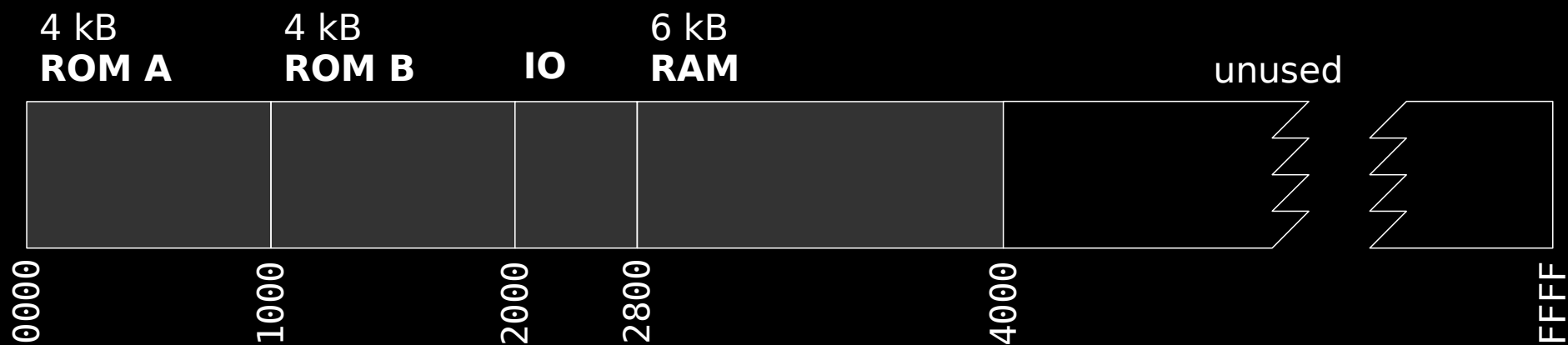










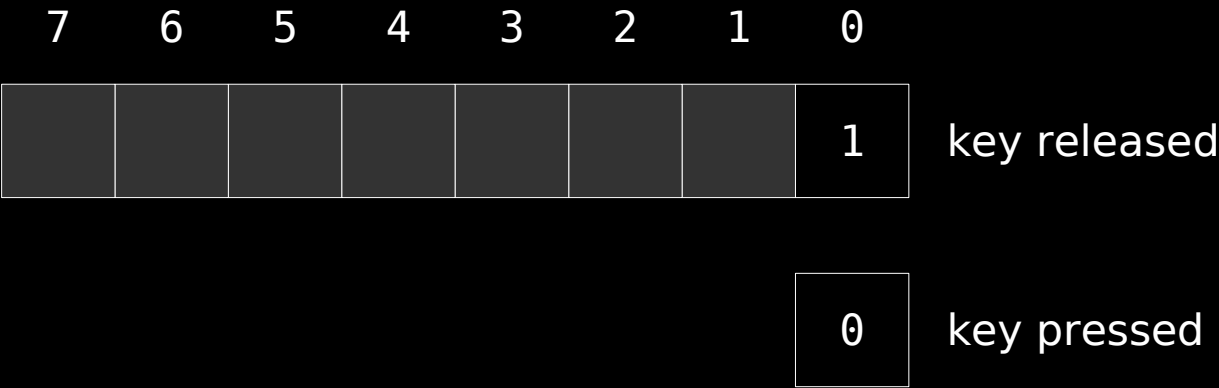


	0		1	2	3	4	5	6	7
0000		tape comparator	A	B	C	D	E	F	G
0008	H		I	J	K	L	M	N	O
0010	P		Q	R	S	T	U	V	W
0018	X		Y	Z	↑	↓	←	→	space
0020	0		1	2	3	4	5	6	7
0028	8		9	;	:	,	=	.	/
0030	return		break	repeat	delete	list	shift		

	0		1	2	3	4	5	6	7
0000		tape comparator	A	B	C	D	E	F	G
0008	H		I	J	K	L	M	N	O
0010	P		Q	R	S	T	U	V	W
0018	X		Y	Z	↑	↓	←	→	space
0020	0		1	2	3	4	5	6	7
0028	8		9	;	:	,	=	.	/
0030	return		break	repeat	delete	list	shift		

[illegible]

	0		1	2	3	4	5	6	7
0000		tape comparator	A	B	C	D	E	F	G
0008	H		I	J	K	L	M	N	O
0010	P		Q	R	S	T	U	V	W
0018	X		Y	Z	↑	↓	←	→	space
0020	0		1	2	3	4	5	6	7
0028	8		9	;	:	,	=	.	/
0030	return		break	repeat	delete	list	shift		



	0	1	2	3	4	5	6	7
0000		A	B	C	D	E	F	G
0008	H	I	J	K	L	M	N	O
0010	P	Q	R	S	T	U	V	W
0018	X	Y	Z	↑	↓	←	→	space
0020	0	1	2	3	4	5	6	7
0028	8	9	;	:	,	=	.	/
0030	return	break	repeat	delete	list	shift		

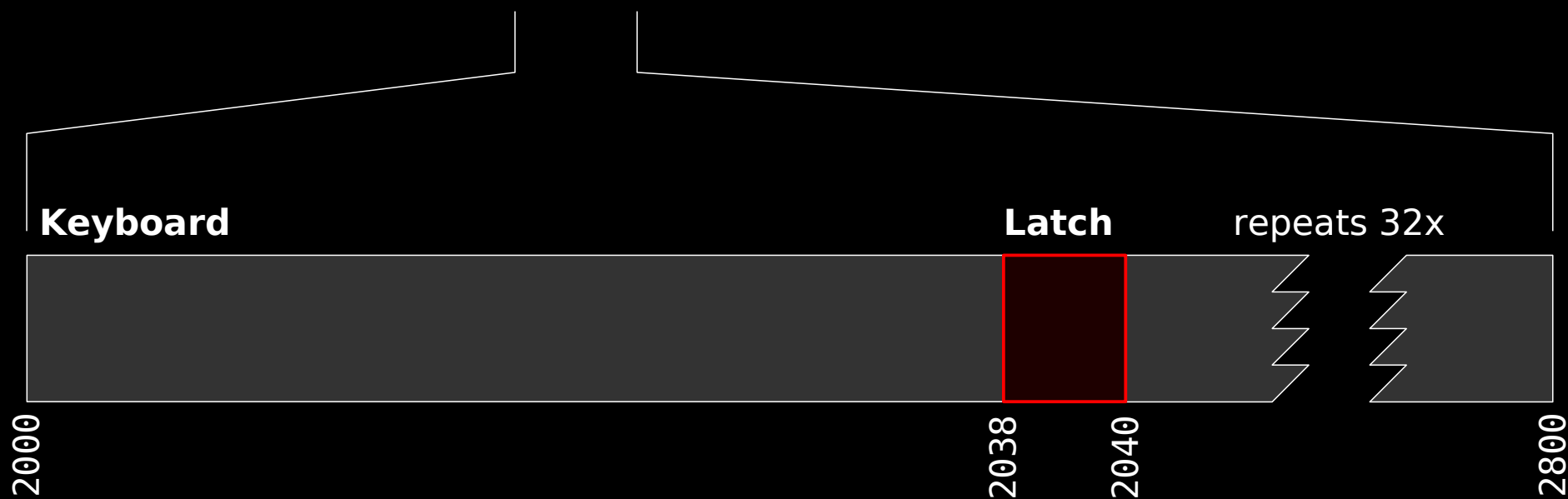
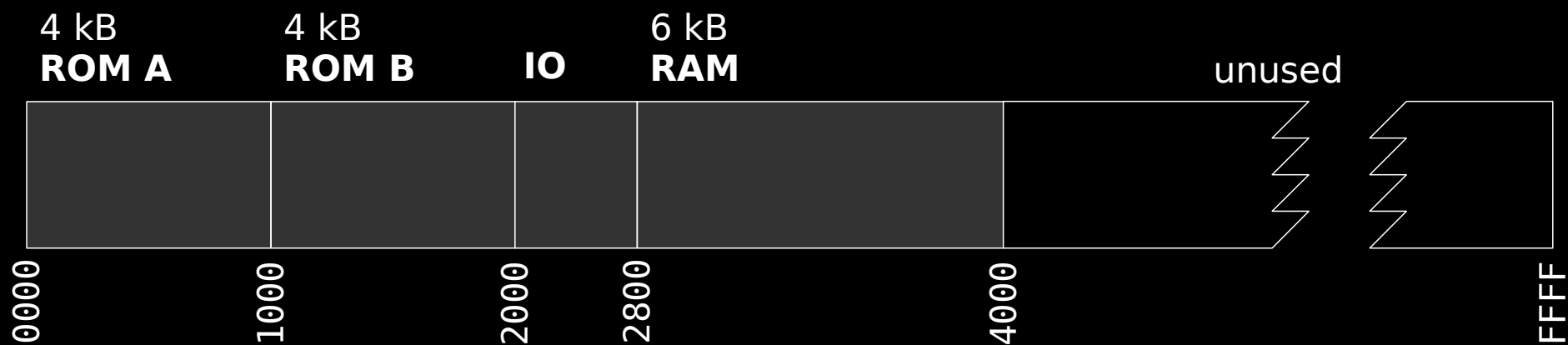
tape comparator

7	6	5	4	3	2	1	0
							1

silence

0

impulse detected

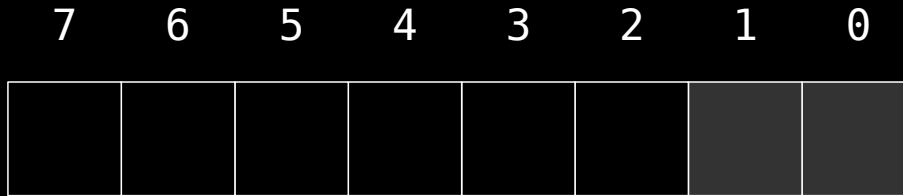


Latch (write only)

7 6 5 4 3 2 1 0

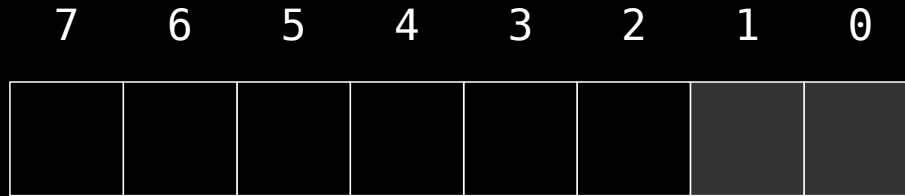
--	--	--	--	--	--	--	--

Latch (write only)



character generator scan line select

Latch (write only)

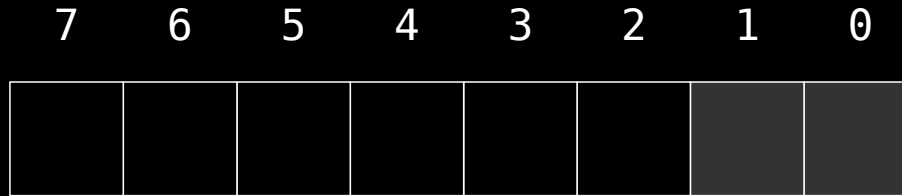


character generator scan line select



cassette port output

Latch (write only)



character generator scan line select



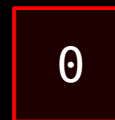
cassette port output



-1.0



0.0

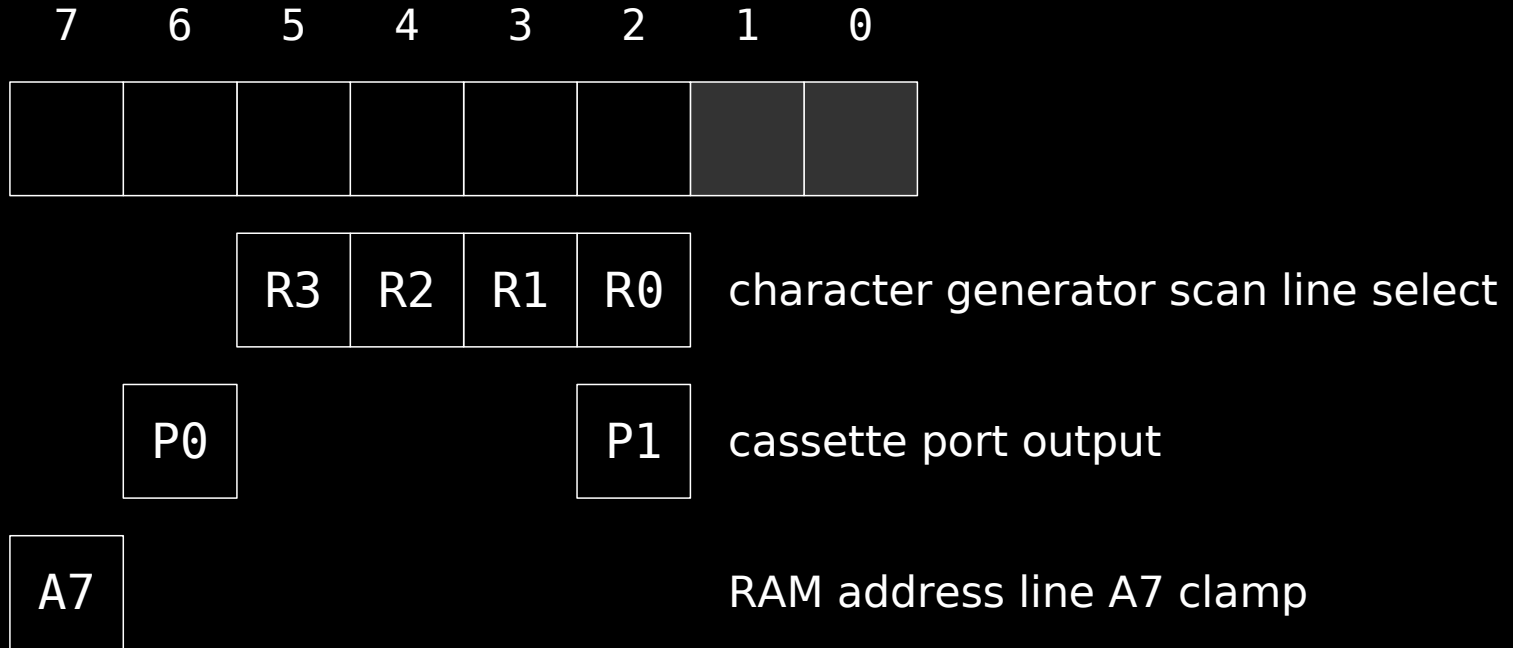


0.0

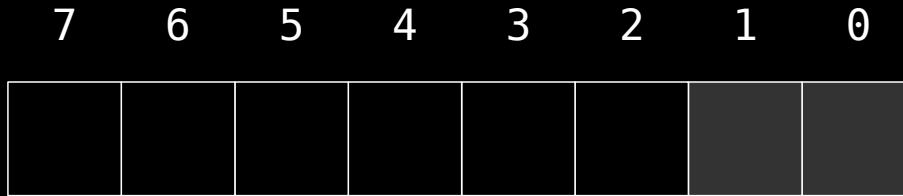


+1.0

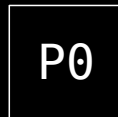
Latch (write only)



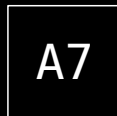
Latch (write only)



character generator scan line select



cassette port output



RAM address line A7 clamp

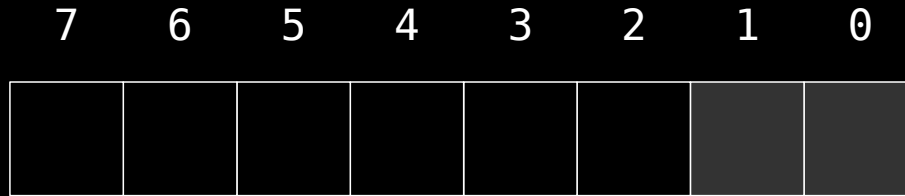


normal operation

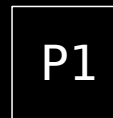


RAM address A7 always set to 1

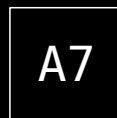
Latch (write only)



character generator scan line select

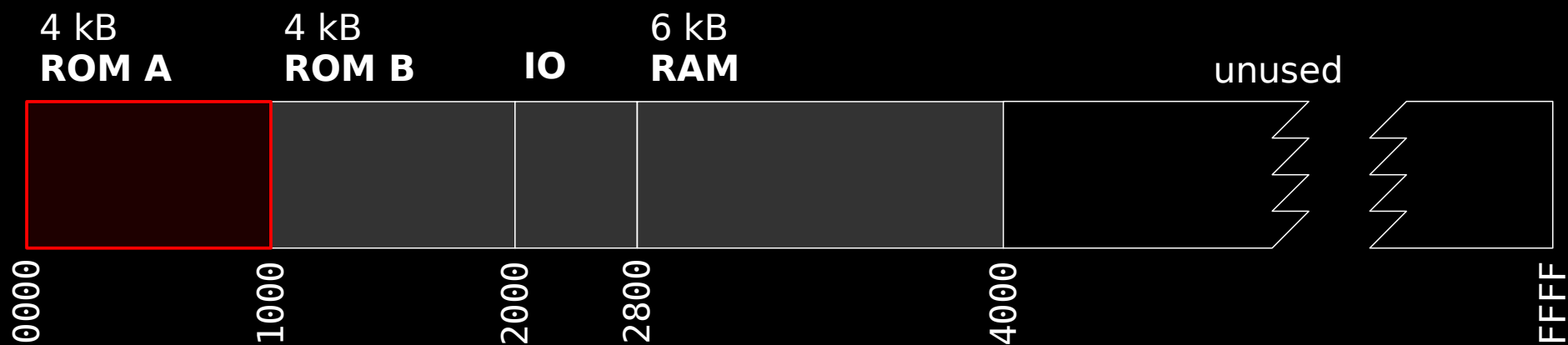


cassette port output



2800 2880 2900 2980 2A00 2A80







```

00010 ;*****
00020 ;*      MALI RACUNAR "GALAKSIJA"      *
00030 ;*      VOJA ANTONIC      *
00040 ;*      03.01.1984.      *
00050 ;*****
00060
00070
00080
1800 00090 XR EQU 1800H
1600 00100 RAM EQU 1600H
28B6 00110 BUF EQU 13B6H+XR
0008 00120 HOR EQU 11
0016 00130 HIRAM EQU RAM<-B&0FFH
2BA8 00140 STACK EQU 13A8H+XR
2C3A 00150 BASIC EQU 143AH+XR
2C36 00160 BASPTR EQU BASIC-4
2C38 00170 BASEND EQU BASIC-2
2AAC 00180 IXPOS EQU 12ACH+XR
00190
0000 F3 00200 DI ; 1
0001 97 00210 SUB A
0002 C3DA03 00220 JP Z18E
0003 DF 00230 ZAREZ RST ZB ; IF (DE)<>"," GO "WHAT"
0006 2C 00240 DEFB ; ELSE RST ZB07
0007 2C 00250 DEFB G08C9-$-1
0008 CDB20A 00260 ZB07 CALL Z24 ; PRETVORI SLEDECI ASCII NIZ U
000B C36D0A 00270 JP ZC0B ; INTEGER I SMESTI GA U HL
000E F3 00280 FAST DI ; FAST FLAG
000F C9 00290 RET
0010 7C 00300 Z32 LD A,H ; CP HL-DE Z & C FLAGS VALID
0011 BA 00310 CP D
0012 C0 00320 RET NZ
0013 7D 00330 LD A,L
0014 BB 00340 CP E
0015 C9 00350 RET
0016 FB 00360 SLOW EI ; SLOW FLAG
0017 C9 00370 RET
0018 E3 00380 ZB EX (SP),HL ; IF ((SP))=(DE) THEN (SP)=(SP)+2
0019 CD0501 00390 CALL Z40 ; ELSE (SP)=((SP)+1)+(SP)+2
001C BE 00400 CP (HL)
001D C39401 00410 JP Z9B
0020 D9 00420 Z16 EXX ; PRINT CHAR IN A ON SCREEN
0021 FE20 00430 CP Z0H
0023 CDB509 00440 CALL ZACB
0026 D9 00450 EXX
0027 C9 00460 RET
002B 210000 00470 HL0 LD HL,0
002B C9 00480 RET
002C CCCC 00490 ZEB9 DEFW 0CCCCCH ; RND CONST
002E CC7E 00500 DEFW 7ECCH
0030 F1 00510 Z48 POP AF ; GO NEXT INSTRUCTION
0031 CD1404 00520 CALL Z8B3
0034 C3BF07 00530 G08C9 JP ZBC9
0037 1C 00540 NUMBER DEFB ZB ; VERZIJA 28
00550
0038 F5 00560 PUSH AF ; INT VIDEO
0039 C5 00570 PUSH BC
003A D5 00580 PUSH DE
003B E5 00590 PUSH HL
003C 21B02B 00600 LD HL,STACK+B
003F 3EC0 00610 LD A,0C0H
0041 96 00620 SUB (HL)
0042 96 00630 SUB (HL)
0043 96 00640 SUB (HL)
0044 5F 00650 LD E,A
0045 7E 00660 LD A,(HL)

```

```

0046 0F 00670 RRCA
0047 0F 00680 RRCA
0048 0F 00690 RRCA
0049 47 00700 LD B,A
004A B7 00710 OR A
004B 2800 00720 JR Z,+2
004D 2805 00730 JR Z,NIJE
004F 35 00740 DEC (HL)
0050 AF 00750 XOR A
0051 D8 00760 SPECL RET C
0052 10FD 00770 DJNZ SPECL
0054 23 00780 NIJE INC HL
0055 77 00790 LD (HL),A
0056 43 00800 LD B,E
0057 217F20 00810 LD HL,207FH
005A 4D 00820 LD C,L
005B 3AA82B 00830 LD A,(STACK)
005E 1F 00840 RRA
005F 3800 00850 JR C,RED
0061 3D 00860 RED DEC A
0062 20FD 00870 JR NZ,RED
0064 1807 00880 JR OVER
0066 F3 00890 DI ; NMI (HARD BREAK)
0067 31A82B 00900 SP1C9 LD SP,STACK
006A C31703 00910 JP Z1C9
006D 0C 00920 OVER INC C
006E 79 00930 LD A,C
006F E608 00940 AND B
0071 0F 00950 RRCA
0072 0F 00960 RRCA
0073 0F 00970 RRCA
0074 F62B 00980 OR 28H
0076 ED47 00990 LD I,A
0078 13 01000 INC DE
0079 110C08 01010 LD DE,B0CH
007C 79 01020 LD A,C
007D 0F 01030 RRCA
007E 0F 01040 RRCA
007F 0F 01050 RRCA
0080 3F 01060 CCF
0081 CB1A 01070 RR D
0083 F61F 01080 OR 1FH
0085 07 01090 RLCA
0086 D640 01100 SUB 40H
0088 0F 01110 RRCA
0089 ED4F 01120 LD R,A
01130
008B 72 01140 LINIJE LD (HL),D
008C 14 01150 INC D
008D 14 01160 INC D
008E 14 01170 INC D
008F 14 01180 INC D
0090 AF 01190 XOR A
0091 37 01200 SCF
0092 1F 01210 RRA
0093 1F 01220 RRA
0094 AA 01230 XOR D
0095 57 01240 LD D,A
0096 61 01250 LD H,C
0097 78 01260 LD A,B
0098 42 01270 BRK DEFM 'BREAK' ; OVAJ STRING JE TRANSP
0099 52
009A 45
009B 41
009C 4B
009D 00 01280 NOP

```

ROM A (4kB)

0x0000

0x0104

0x09b5

0x0a45

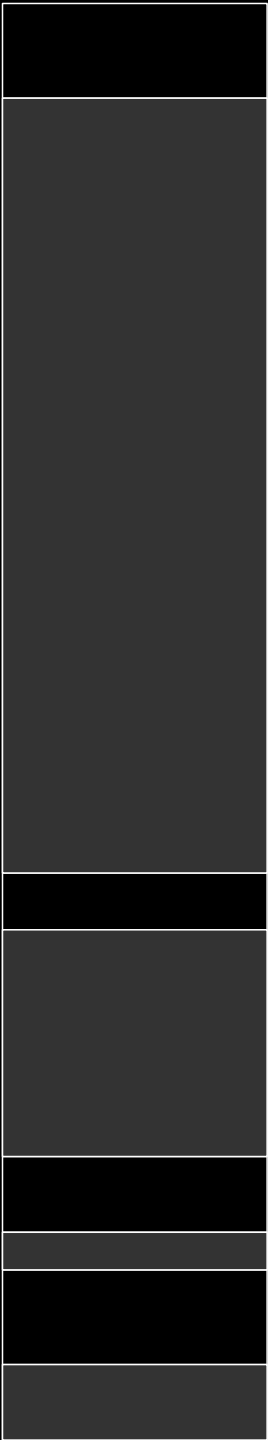
0x0ce6

0x0d9a

0x0e30

0x0f0f

0x1000



0x0000

0x0104

0x09b5

0x0a45

0x0ce6

0x0d9a

0x0e30

0x0f0f

0x1000

Reset vectors

ROM A (4kB)



0x0000

0x0104

0x09b5

0x0a45

0x0ce6

0x0d9a

0x0e30

0x0f0f

0x1000

Reset vectors

Video driver

ROM A (4kB)



0x0000

Reset vectors

0x0104

Video driver

ROM A (4kB)

BASIC interpreter

0x09b5

0x0a45

0x0ce6

0x0d9a

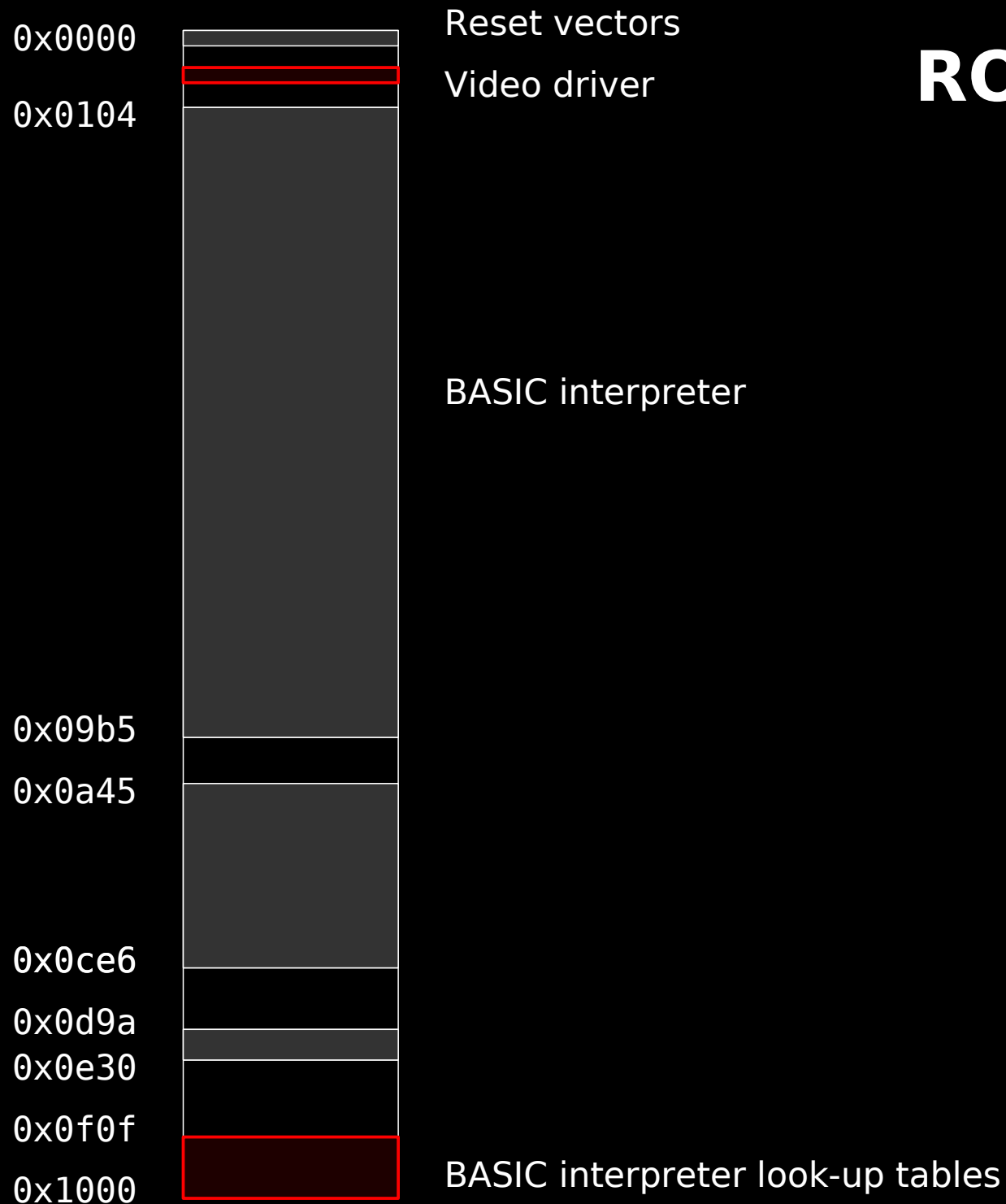
0x0e30

0x0f0f

0x1000



ROM A (4kB)



```

inc d      ;008d    4
inc d      ;008e    4
inc d      ;008f    4
xor a      ;0090    4
scf        ;0091    4
rra        ;0092    4
rra        ;0093    4
xor d      ;0094    4
ld d,a     ;0095    4
ld h,c     ;0096    4
ld a,b     ;0097    4

```

;; BREAK_STRING

10098h:

; This part of the line drawing code also serves as a string "BREAK" for the
; basic interpreter.

ld b,d	;0098	4	B
ld d,d	;0099	4	R
ld b,l	;009a	4	E
ld b,c	;009b	4	A
ld c,e	;009c	4	K
nop	;009d	4	\0

```

ld b,a     ;009e    4
ld c,h     ;009f    4

```

; This part of the line drawing code also serves as a constant 1.0 in four-byte
; floating point format.

;; FP_ONE

100a0h:

```

nop        ;00a0    4    00h  Mantissa: 800000h
nop        ;00a1    4    00h
add a,b     ;00a2    4    80h  Sign bit: 00h

```

Command line only

LIST
RUN
NEW
SAVE
OLD
EDIT

Command line only

LIST
RUN
NEW
SAVE
OLD
EDIT

Functions

RND
MEM
KEY
BYTE
WORD
PTR
VAL
EQ
INT
&
USR
DOT

Command line only

LIST
RUN
NEW
SAVE
OLD
EDIT

Functions

RND
MEM
KEY
BYTE
WORD
PTR
VAL
EQ
INT
&
USR
DOT

Procedures

NEXT
INPUT
IF
GOTO
CALL
UNDOT
RET
TAKE
!

FOR
PRINT
DOT
ELSE
BYTE
WORD
ARR\$
STOP
HOME

Command line only

LIST
RUN
NEW
SAVE
OLD
EDIT

Functions

RND
MEM
KEY
BYTE
WORD
PTR
VAL
EQ
INT
&
USR
DOT

Procedures

NEXT
INPUT
IF
GOTO
CALL
UNDOT
RET
TAKE
!

FOR
PRINT
DOT
ELSE
BYTE
WORD
ARR\$
STOP
HOME

Error messages

WHAT? (parsing error)
HOW? (invalid argument)
SORRY (out of memory)

Command line only

LIST
RUN
NEW
SAVE
OLD
EDIT

Functions

RND
MEM
KEY
BYTE
WORD
PTR
VAL
EQ
INT
&
USR
DOT

Extendable

g_basic_link @ 0x2ba9

Procedures

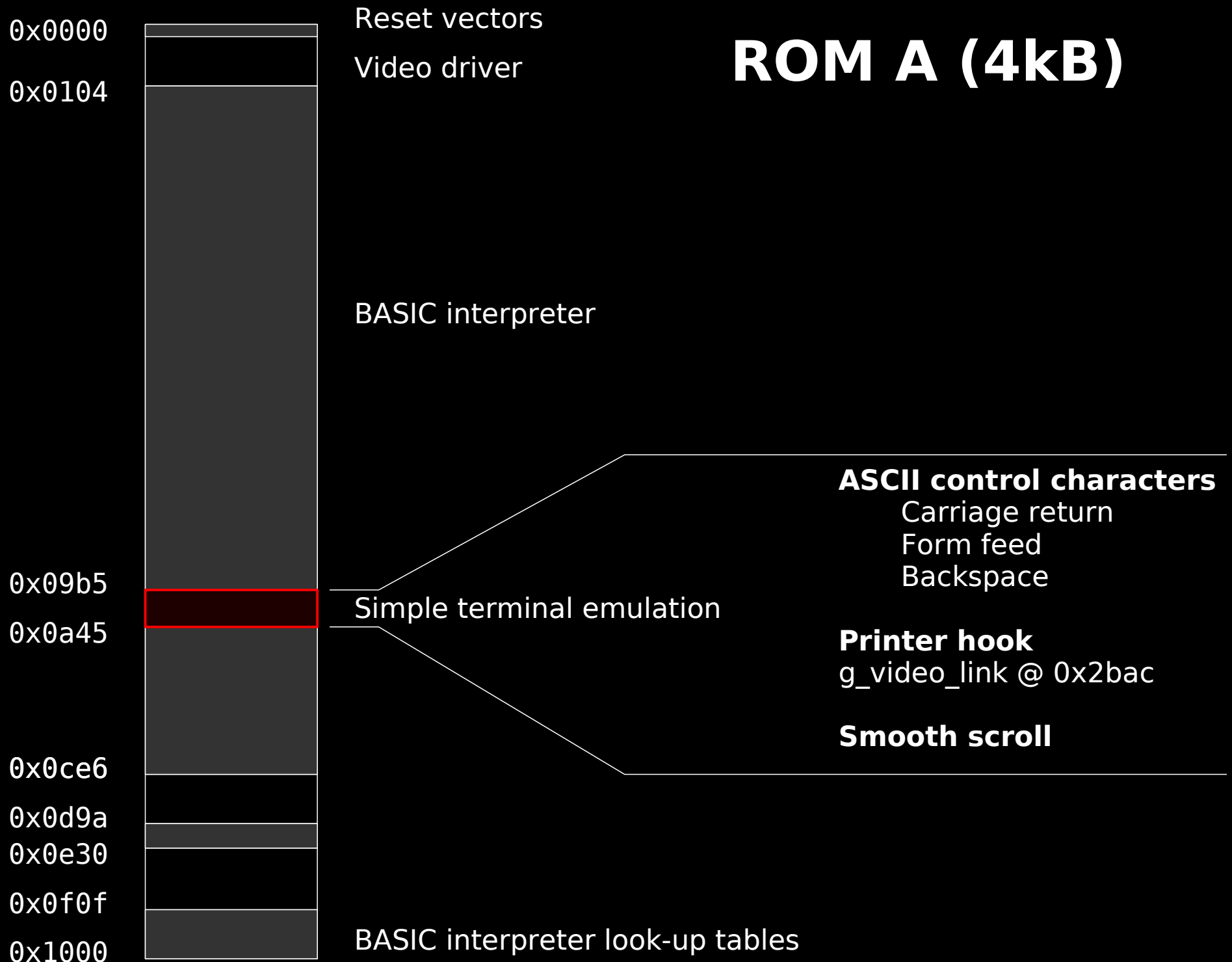
NEXT
INPUT
IF
GOTO
CALL
UNDOT
RET
TAKE
!

FOR
PRINT
DOT
ELSE
BYTE
WORD
ARR\$
STOP
HOME

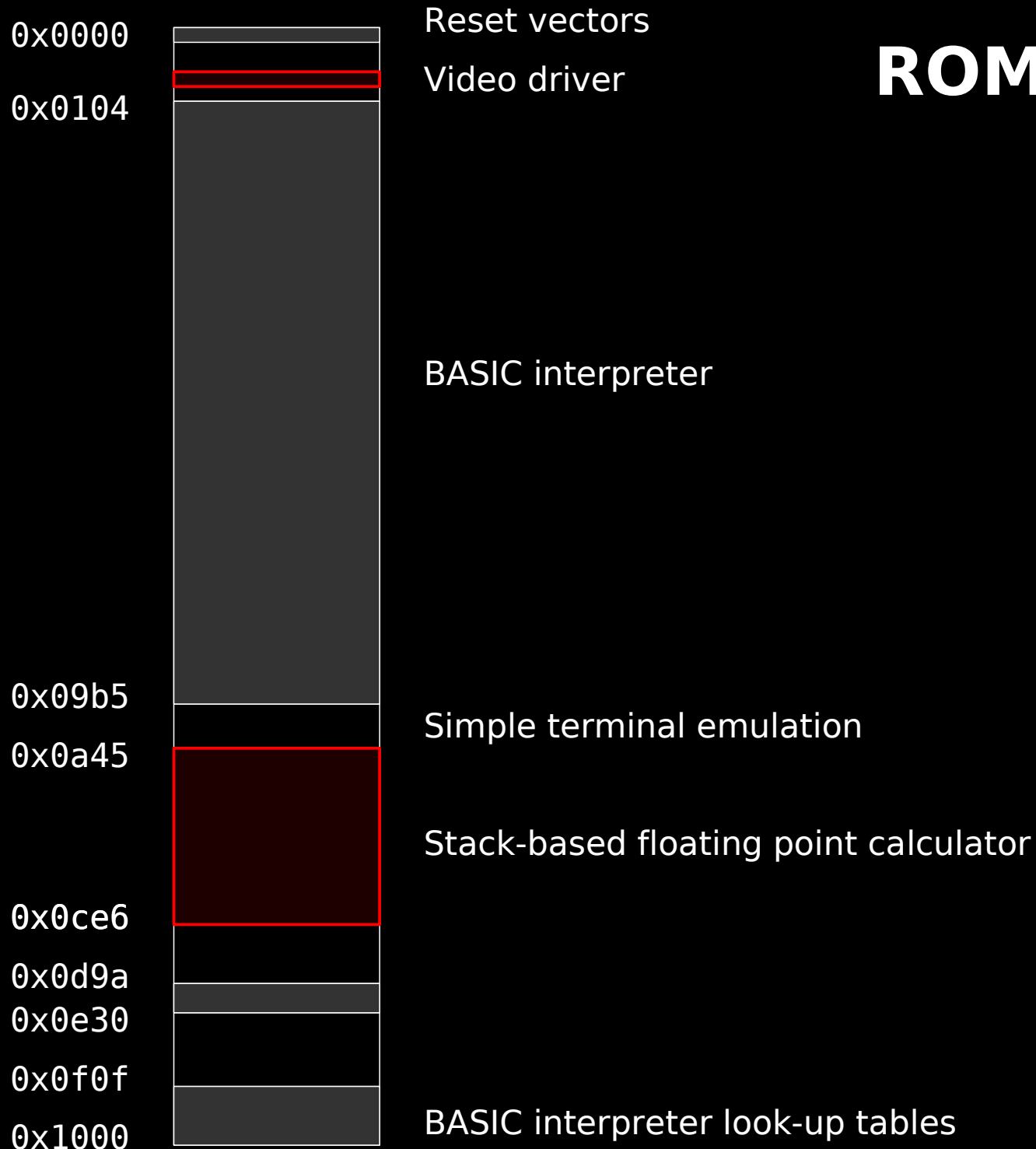
Error messages

WHAT? (parsing error)
HOW? (invalid argument)
SORRY (out of memory)

ROM A (4kB)



ROM A (4kB)



ROM A (4kB)

0x0000

Reset vectors

0x0104

Video driver

BASIC interpreter

0x09b5

Simple terminal emulation

0x0a45

40-bit floating point format

MMMMMMMM MMMMMMMM 1MMMMMMM EEEEEEEE S0000000

0x0ce6

32-bit floating point format

MMMMMMMM MMMMMMMM EMMMMMMM SEEEEEEE

0x0d9a

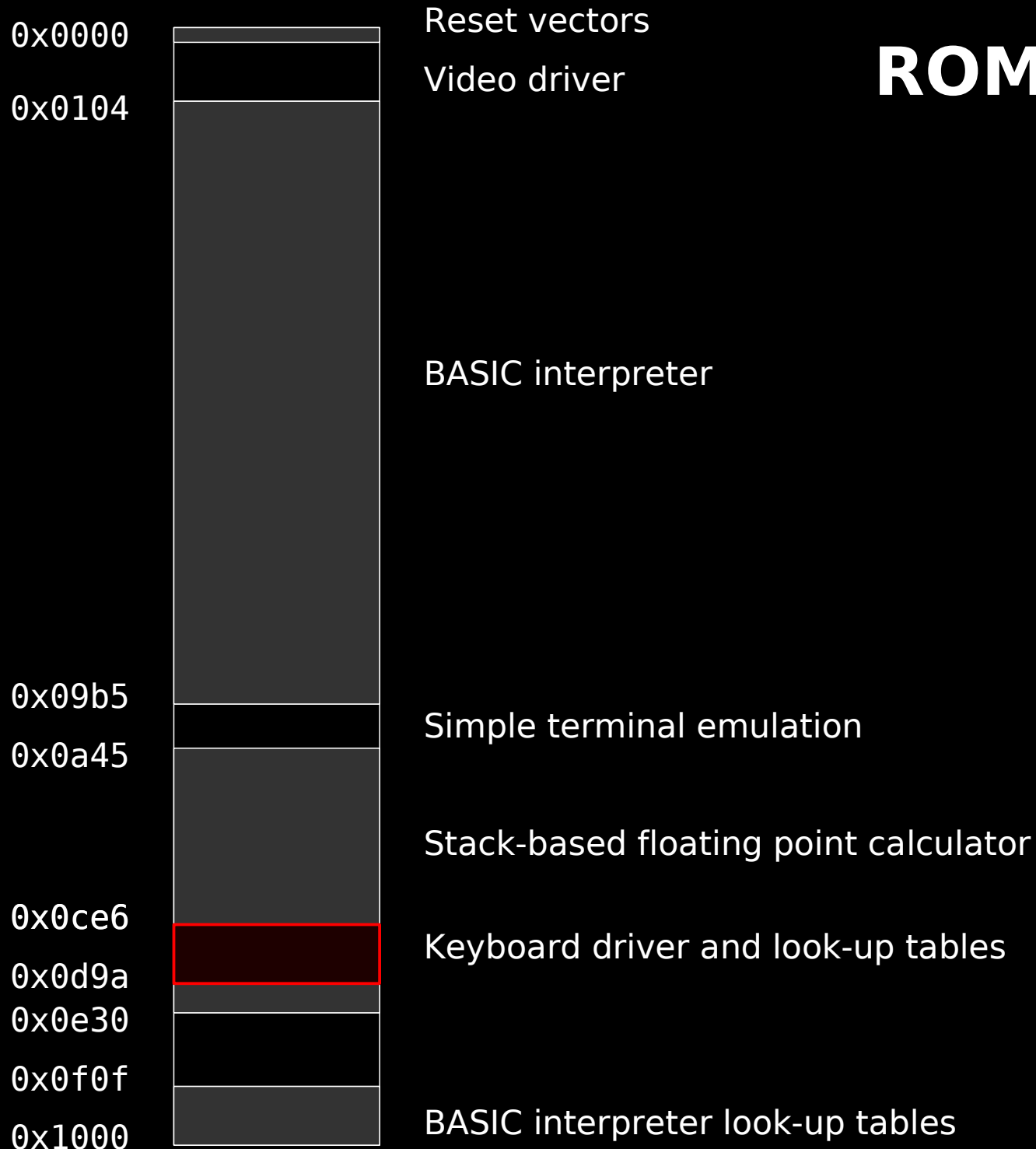
0x0e30

0x0f0f

23-bit mantissa, 8-bit exponent

0x1000

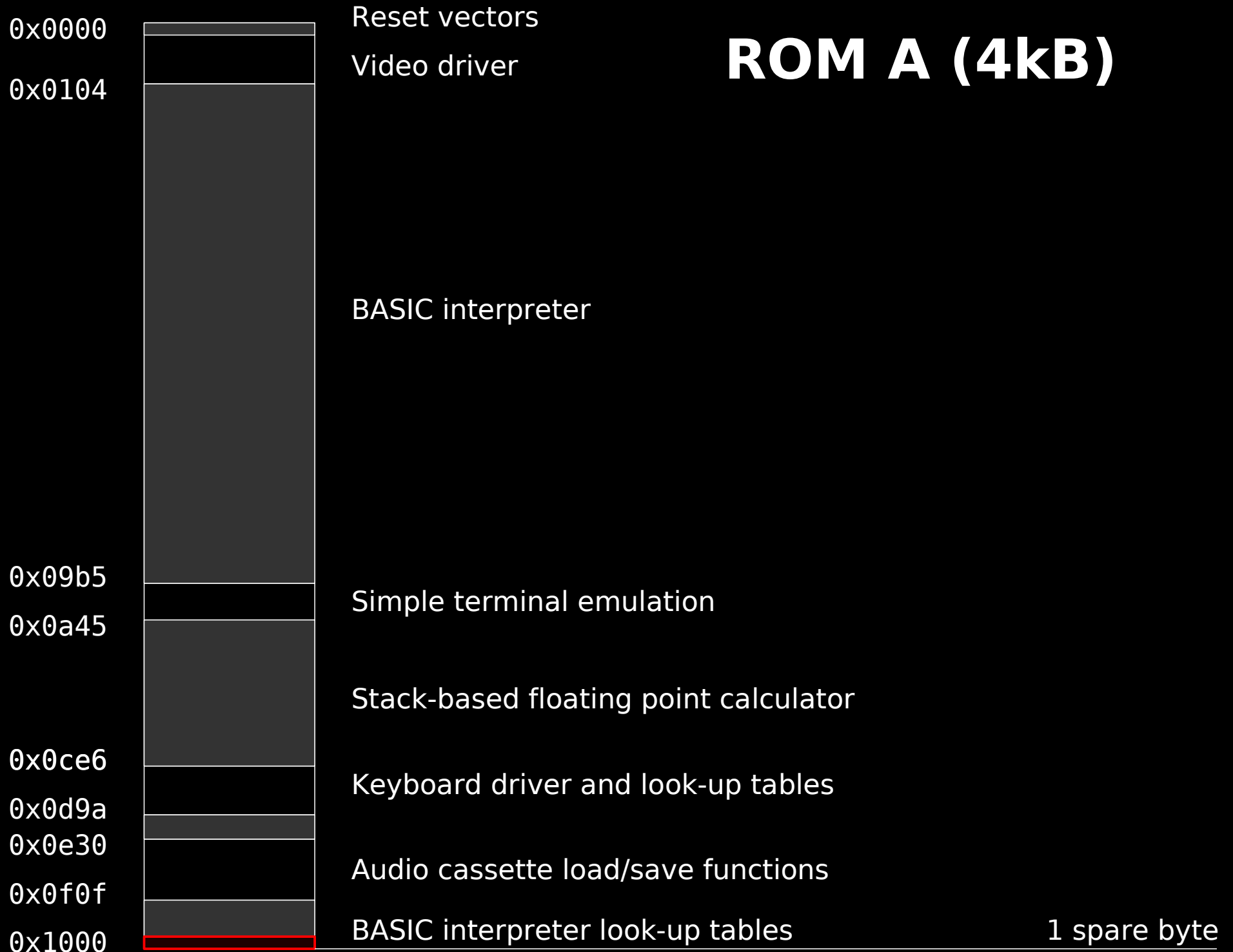
ROM A (4kB)

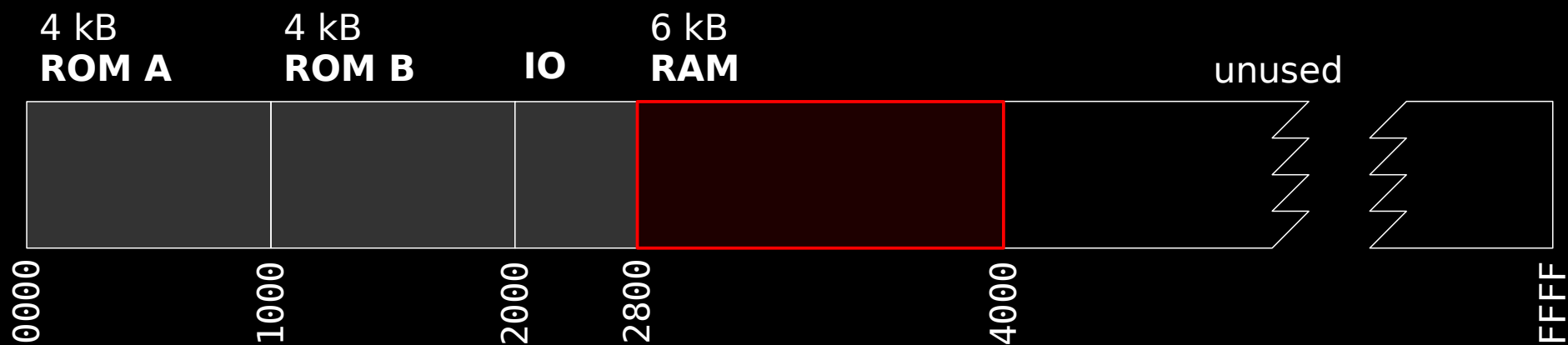


ROM A (4kB)

0x0000	Reset vectors
0x0104	Video driver
	BASIC interpreter
0x09b5	
0x0a45	Simple terminal emulation
	Stack-based floating point calculator
0x0ce6	
0x0d9a	Keyboard driver and look-up tables
0x0e30	
0x0f0f	Audio cassette load/save functions
0x1000	BASIC interpreter look-up tables

ROM A (4kB)





0x2800

RAM (System area)

0x2a00

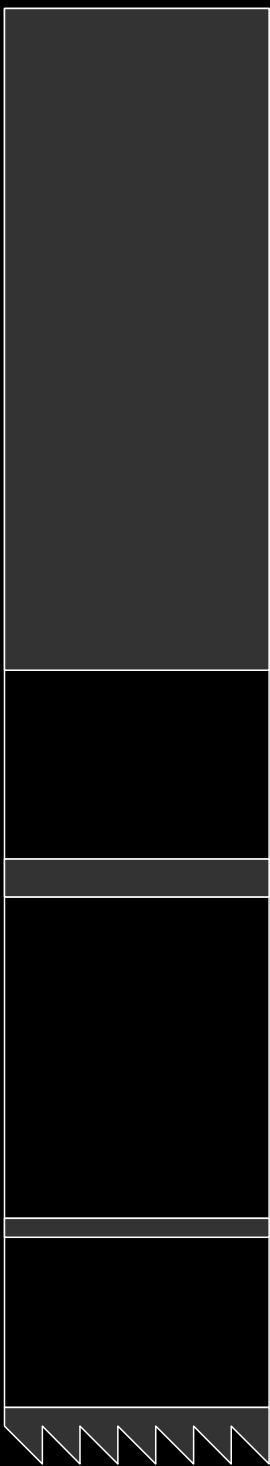
0x2a91

0x2aac

0x2ba8

0x2c36

0x2c3a



0x2800

RAM (System area)

Video frame buffer

32 x 16 characters
512 bytes

0x2a00

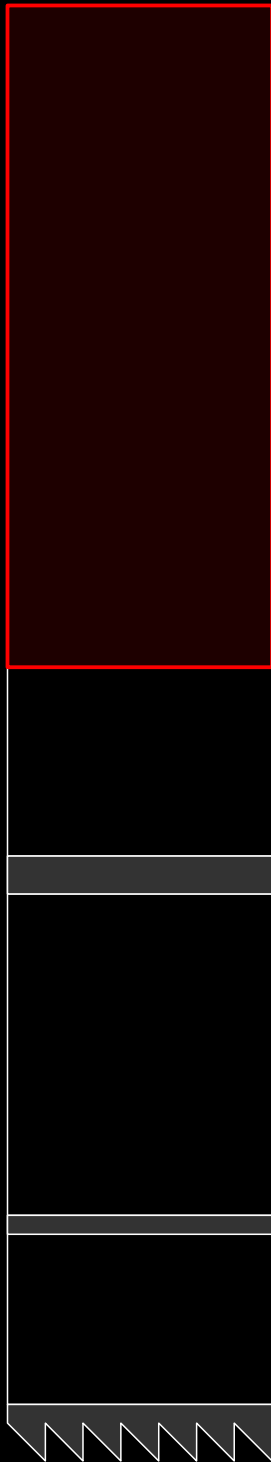
0x2a91

0x2aac

0x2ba8

0x2c36

0x2c3a



0x2800

RAM (System area)

Video frame buffer

32 x 16 characters
512 bytes

0x2a00

A B C D E F G H I J K ... Z
26 numerical variables
4 bytes each

BASIC variables

0x2a91

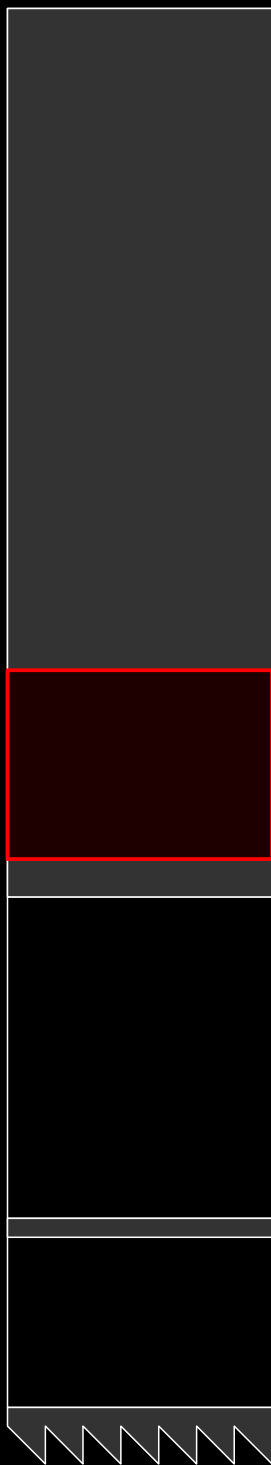
\$X \$Y
2 string variables
16 bytes each

0x2aac

0x2ba8

0x2c36

0x2c3a



0x2800

RAM (System area)

Video frame buffer

32 x 16 characters
512 bytes

0x2a00

BASIC variables

A B C D E F G H I J K ... Z
26 numerical variables
4 bytes each

0x2a91

\$X \$Y
2 string variables
16 bytes each

0x2aac

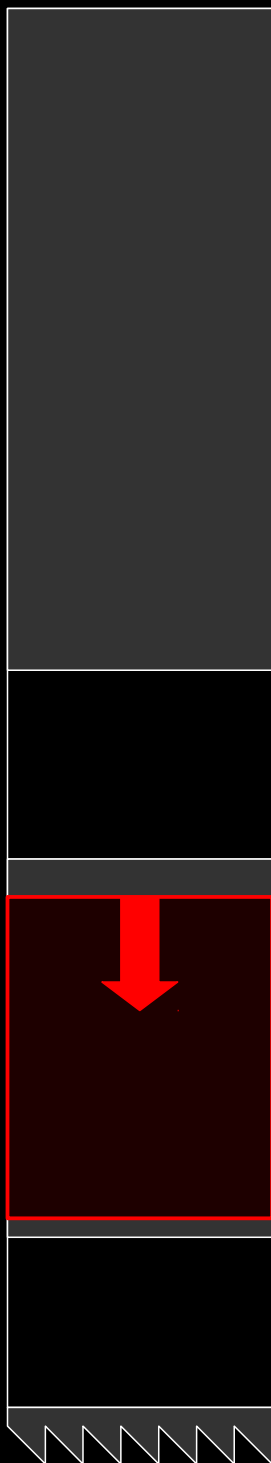
Arithmetic
stack

63 positions
4 bytes each

0x2ba8

0x2c36

0x2c3a



0x2800

RAM (System area)

Video frame buffer

32 x 16 characters
512 bytes

0x2a00

BASIC variables

A B C D E F G H I J K ... Z
26 numerical variables
4 bytes each

0x2a91

\$X \$Y

0x2aac

2 string variables
16 bytes each

Arithmetic
stack

63 positions
4 bytes each

CPU stack

0x2ba8

0x2c36

0x2c3a



0x2800

RAM (System area)

Video frame buffer

32 x 16 characters
512 bytes

0x2a00

BASIC variables

A B C D E F G H I J K ... Z
26 numerical variables
4 bytes each

0x2a91

0x2aac

\$X \$Y
2 string variables
16 bytes each

Arithmetic
stack

CPU stack

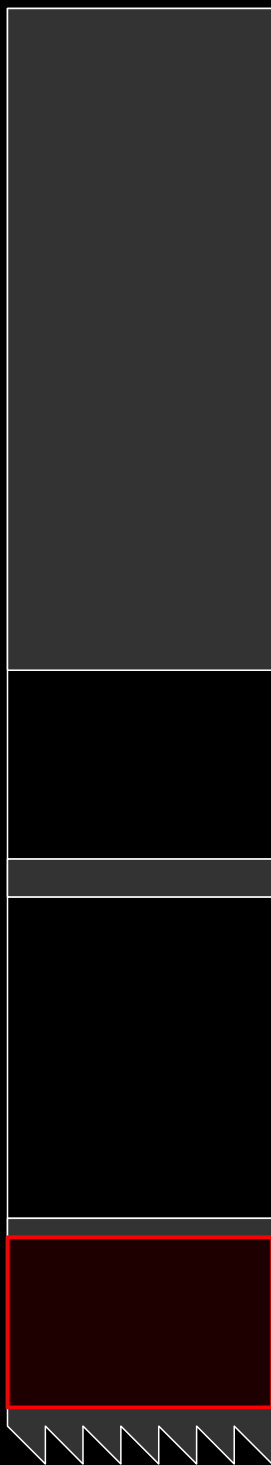
63 positions
4 bytes each

0x2ba8

Multi-purpose buffer

0x2c36

0x2c3a



0x2800

RAM (System area)

Video frame buffer

32 x 16 characters
512 bytes

0x2a00

BASIC variables

A B C D E F G H I J K ... Z
26 numerical variables
4 bytes each

0x2a91

\$X \$Y

0x2aac

2 string variables
16 bytes each

Arithmetic
stack

63 positions
4 bytes each

CPU stack

0x2ba8

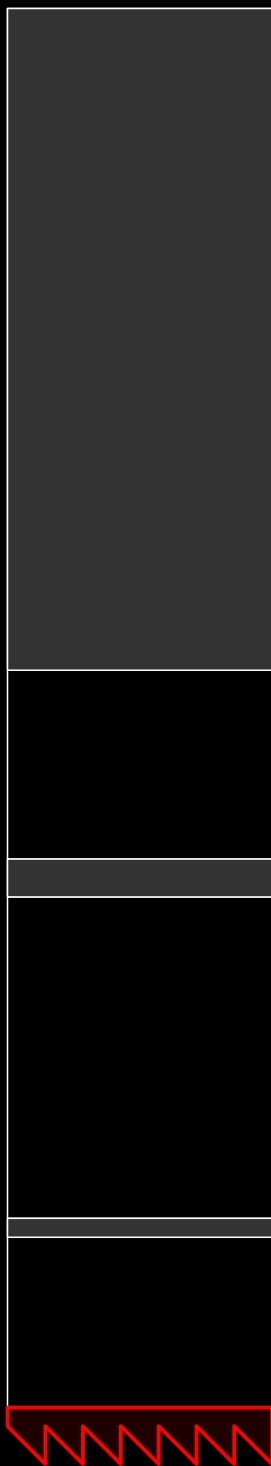
Multi-purpose buffer

0x2c36

0x2c3a

User space

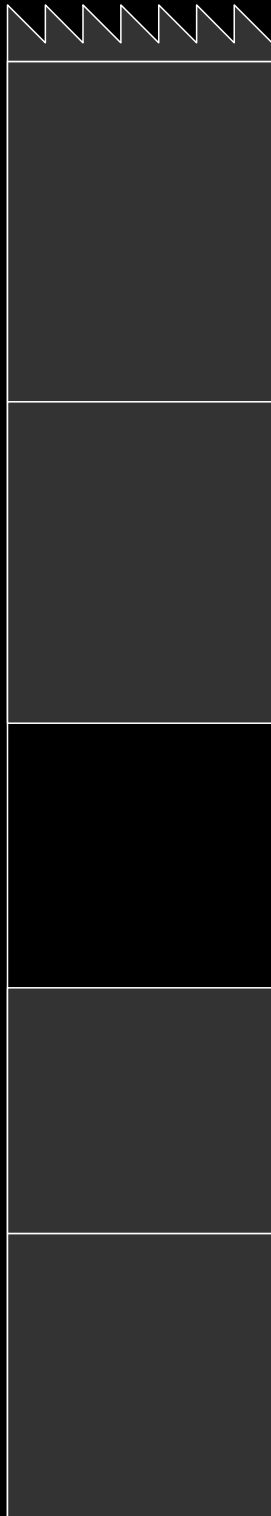
5062 bytes



0x2c3a

RAM (User space)

0x3fff

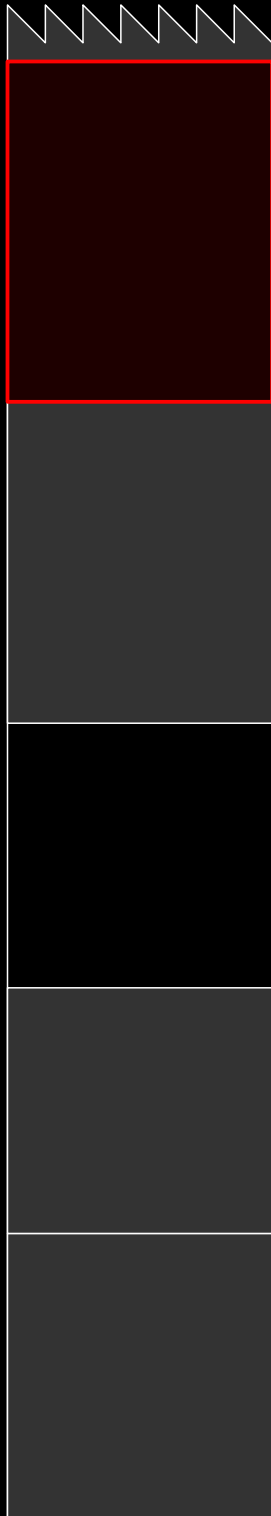


0x2c3a

RAM (User space)

Machine code program

0x3fff



0x2c3a

RAM (User space)

Machine code program

g_basic_start

BASIC program

g_basic_end



0x3fff

0x2c3a

RAM (User space)

Machine code program

g_basic_start

BASIC program

g_basic_end



BASIC A(I) numerical array

0x3fff

0x2c3a

RAM (User space)

Machine code program

g_basic_start

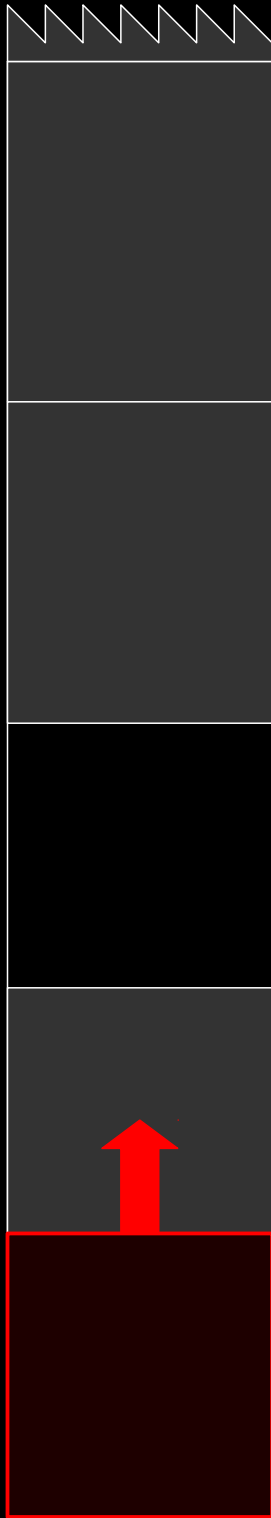
BASIC program

g_basic_end

BASIC A(I) numerical array

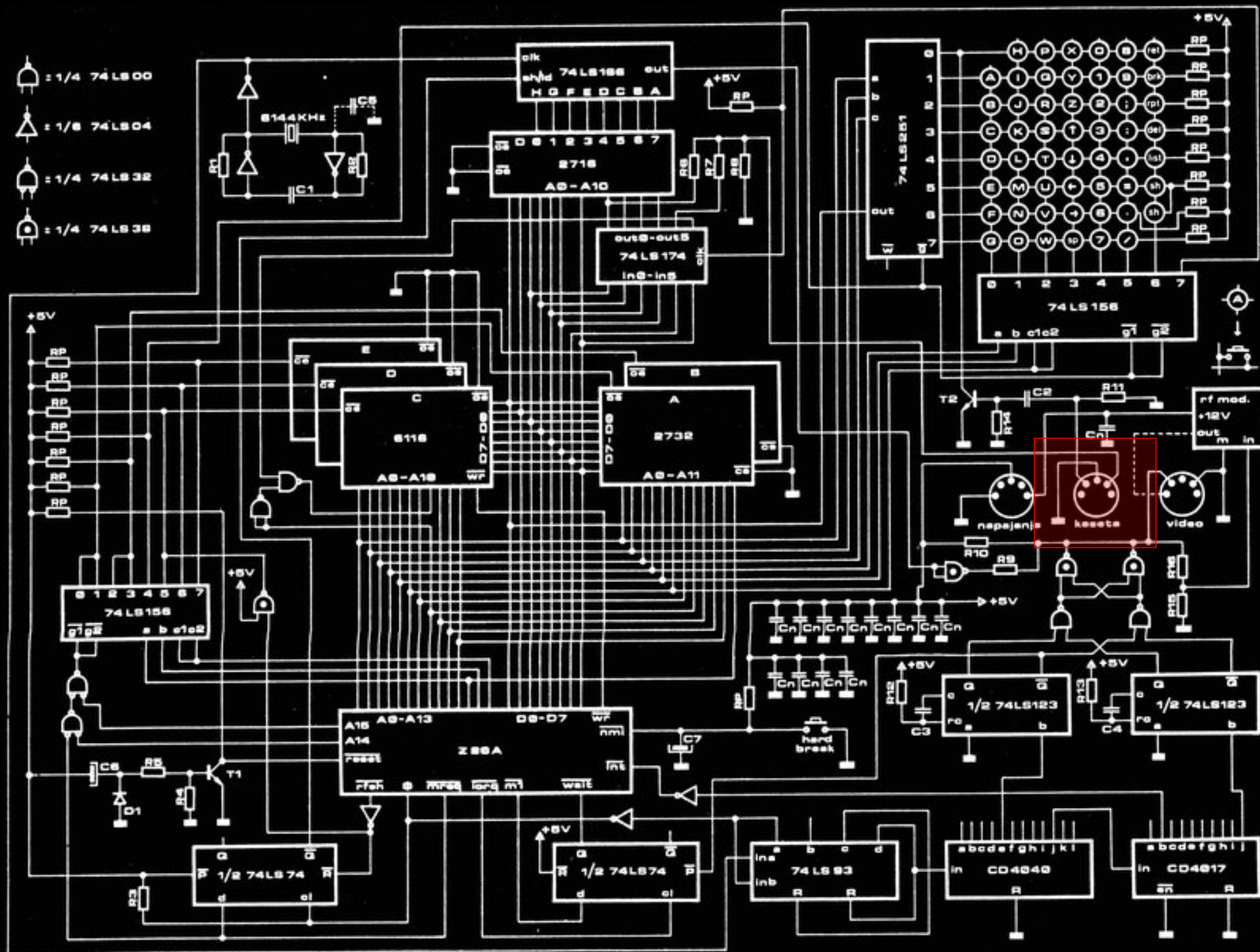
BASIC X\$(I) string array

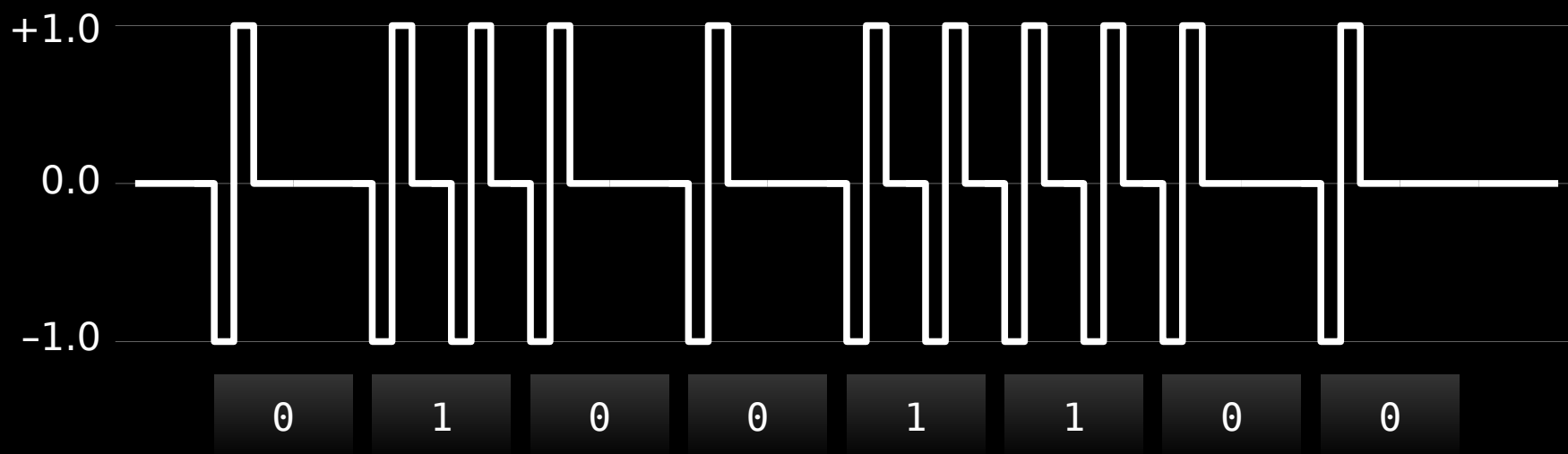
0x3fff

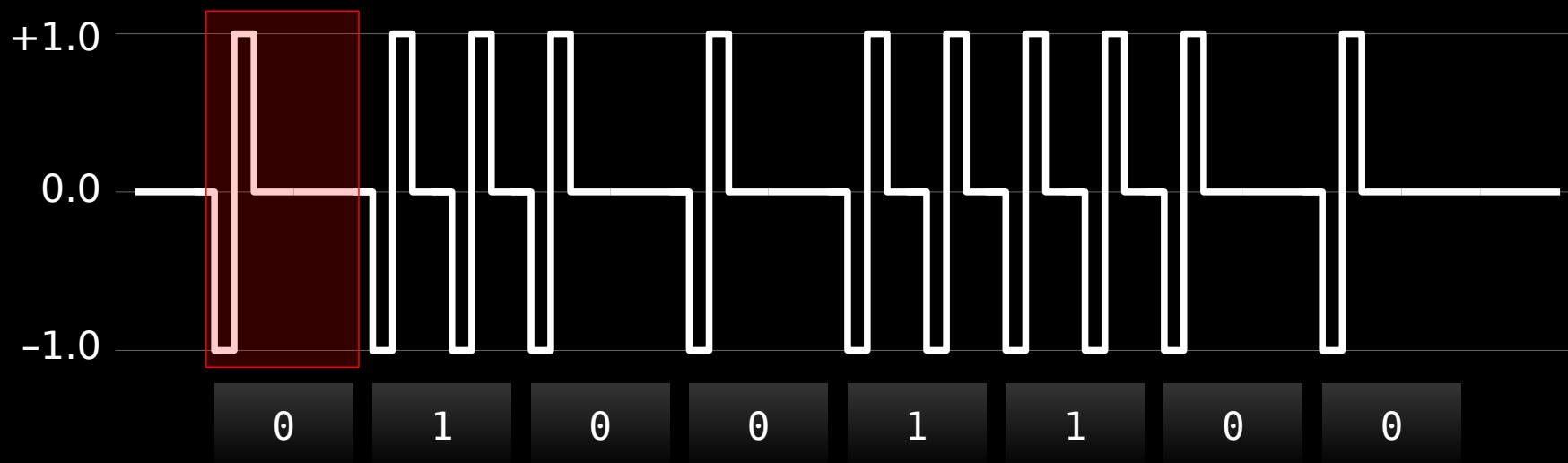


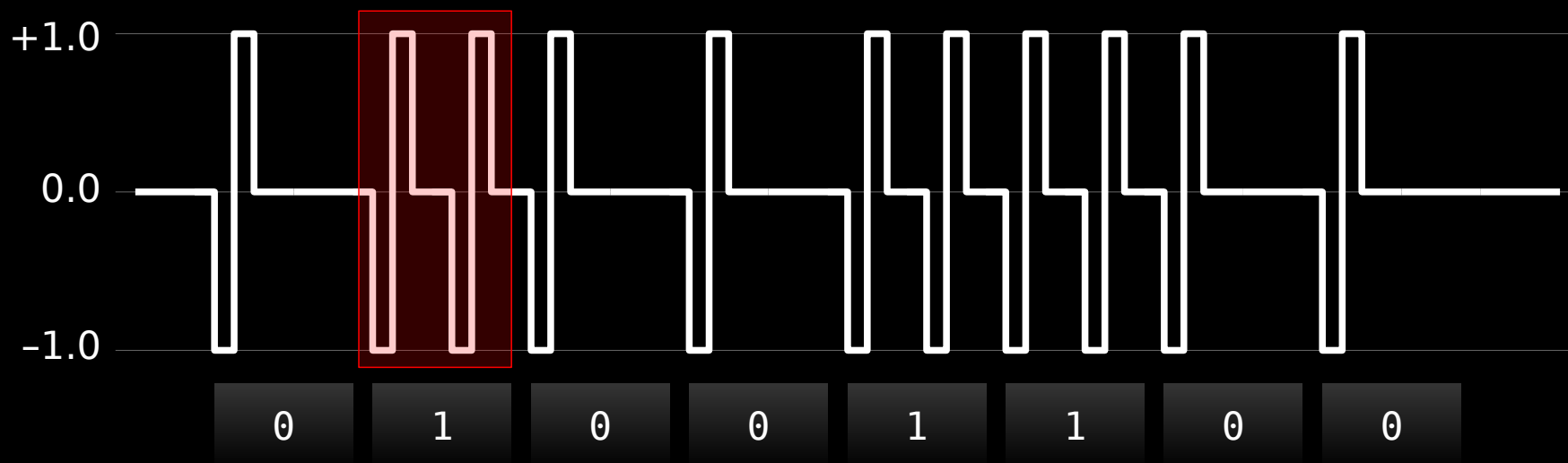
Other useful POKEs

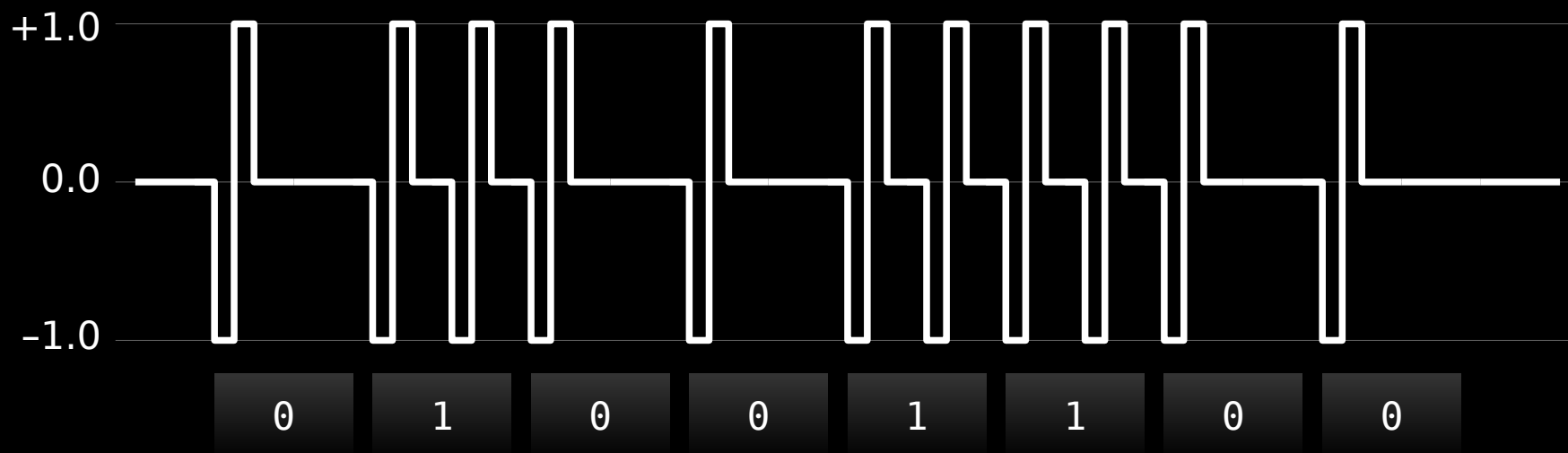
- **Horizontal, vertical screen position**
g_horiz_pos @ 0x2ba8
g_scroll_cnt @ 0x2bb0
- **Start of BASIC program**
g_basic_start @ 0x2c36
- **End of BASIC program**
g_basic_end @ 0x2c38
- **IX register** – arithmetic stack pointer
- **IY register** – video interrupt hook











330 bit/s

a5
36
2c
8d
33

33

0

a5

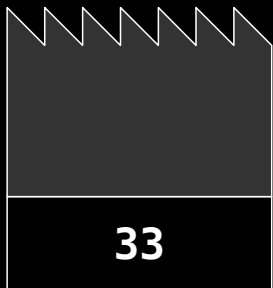
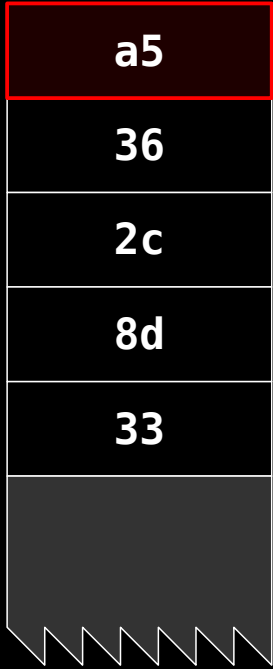
Sync byte

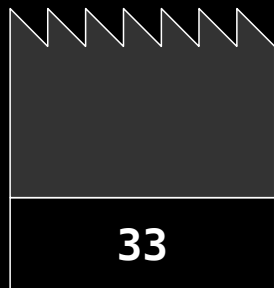
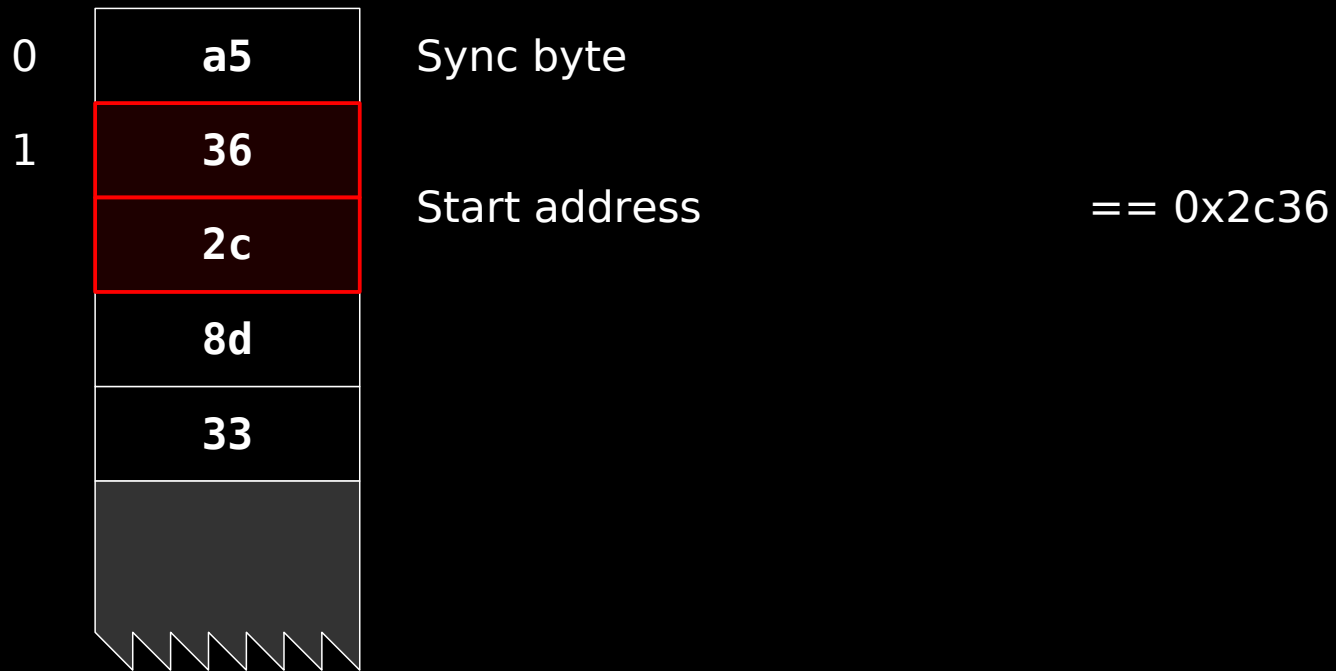
36

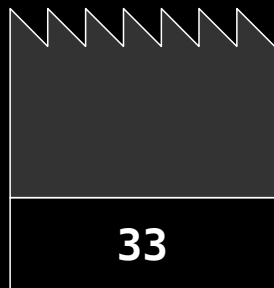
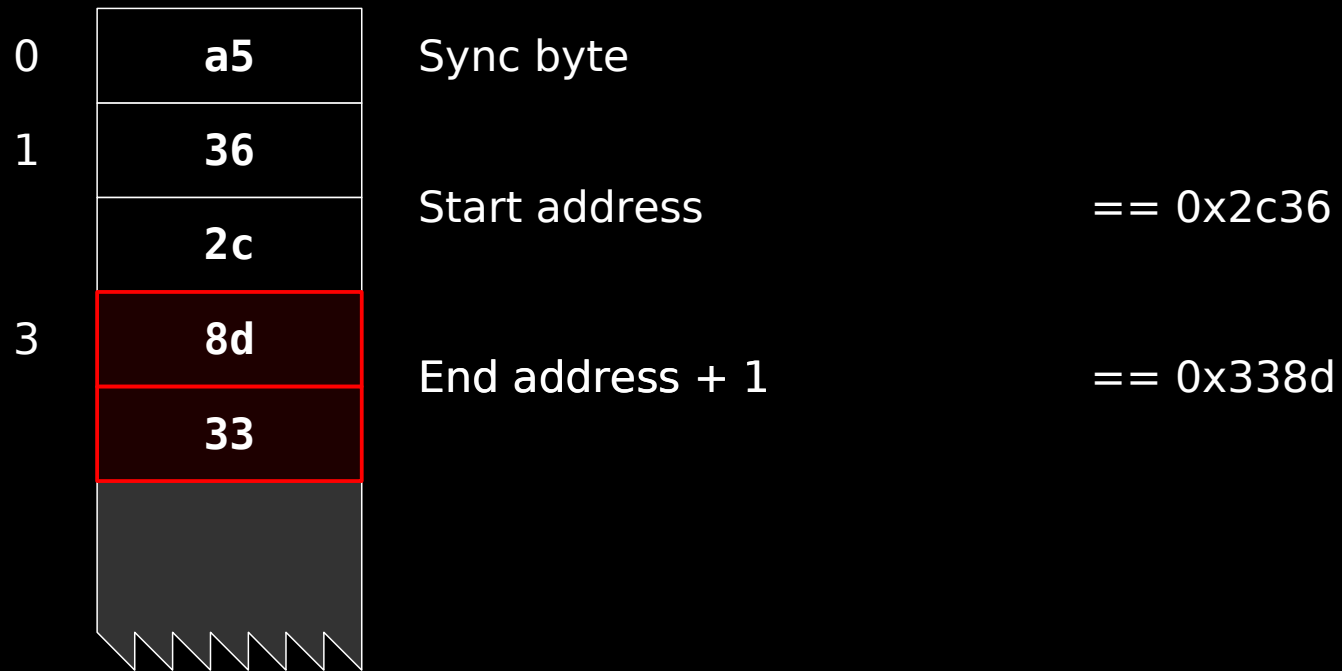
2c

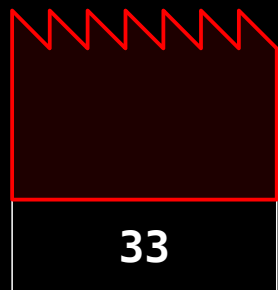
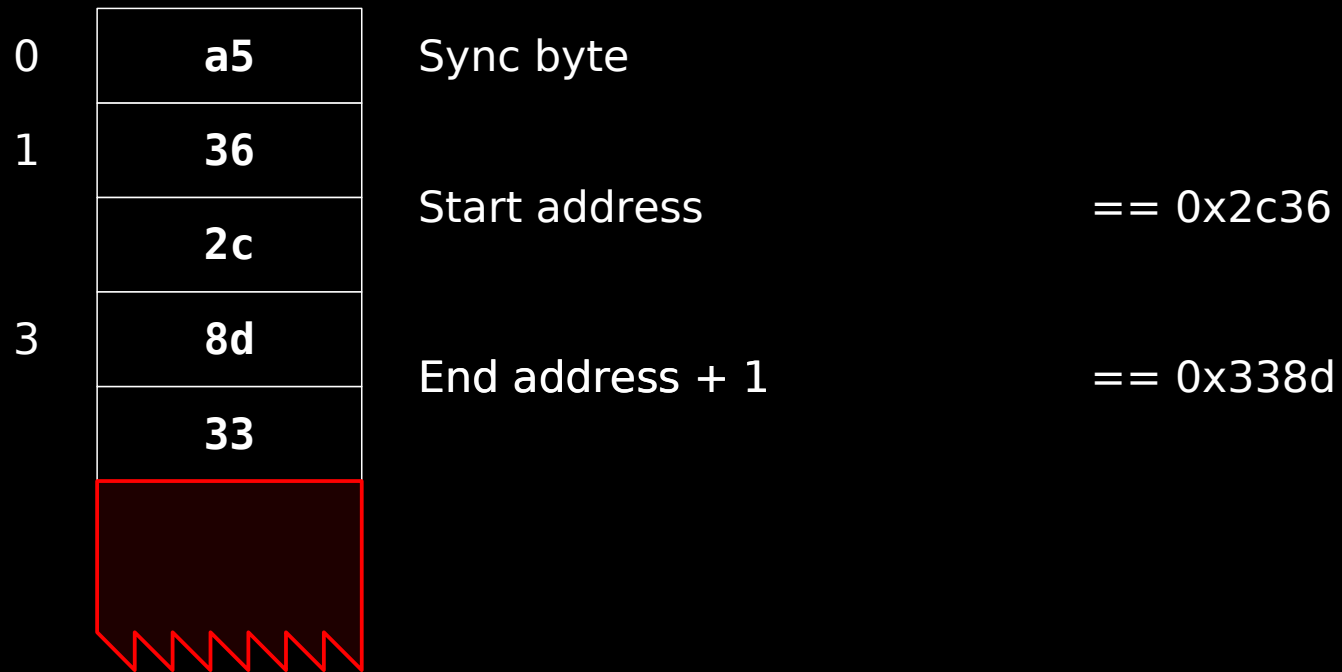
8d

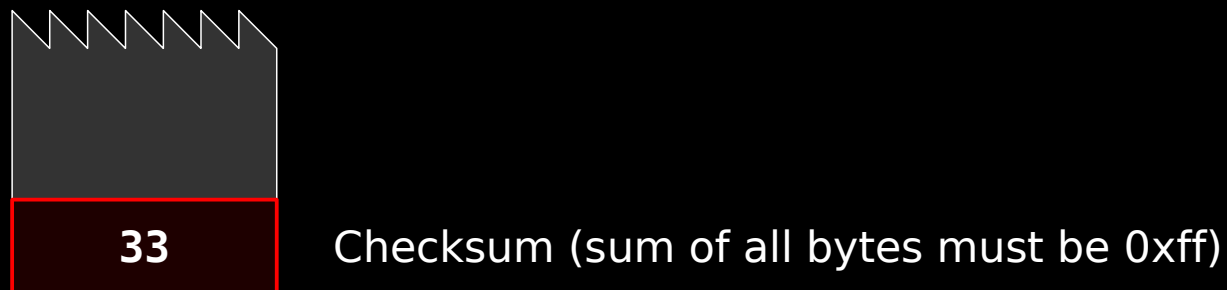
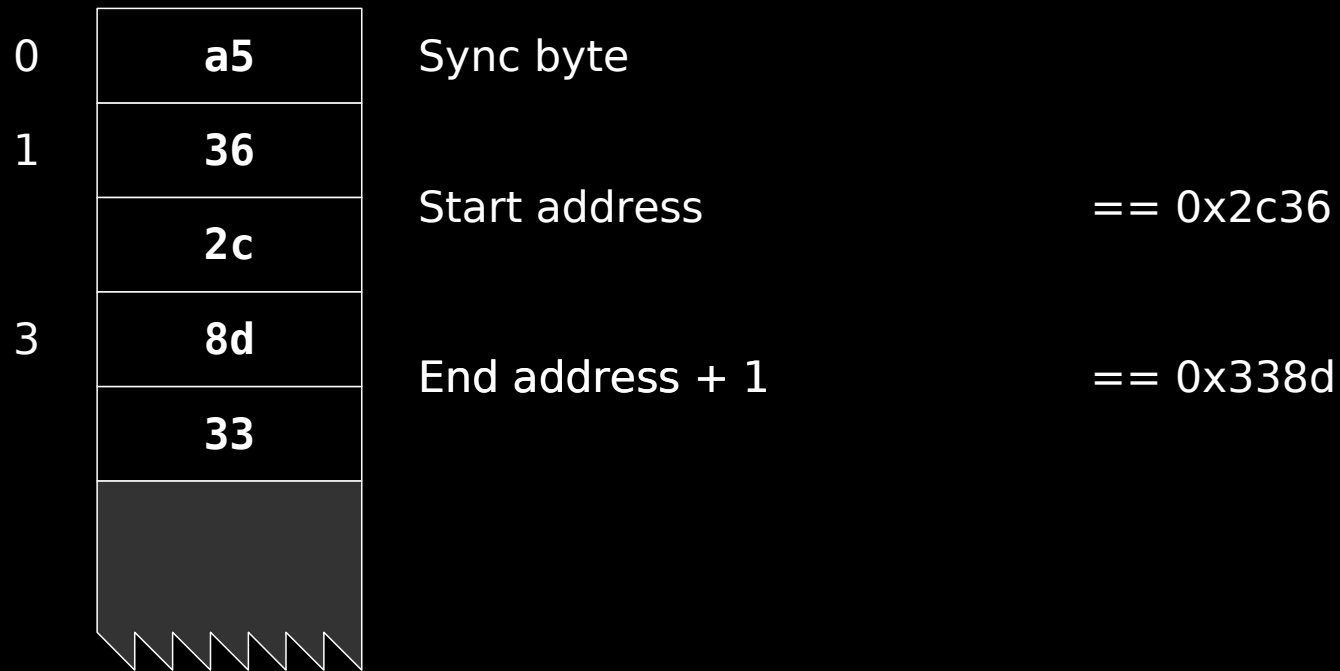
33

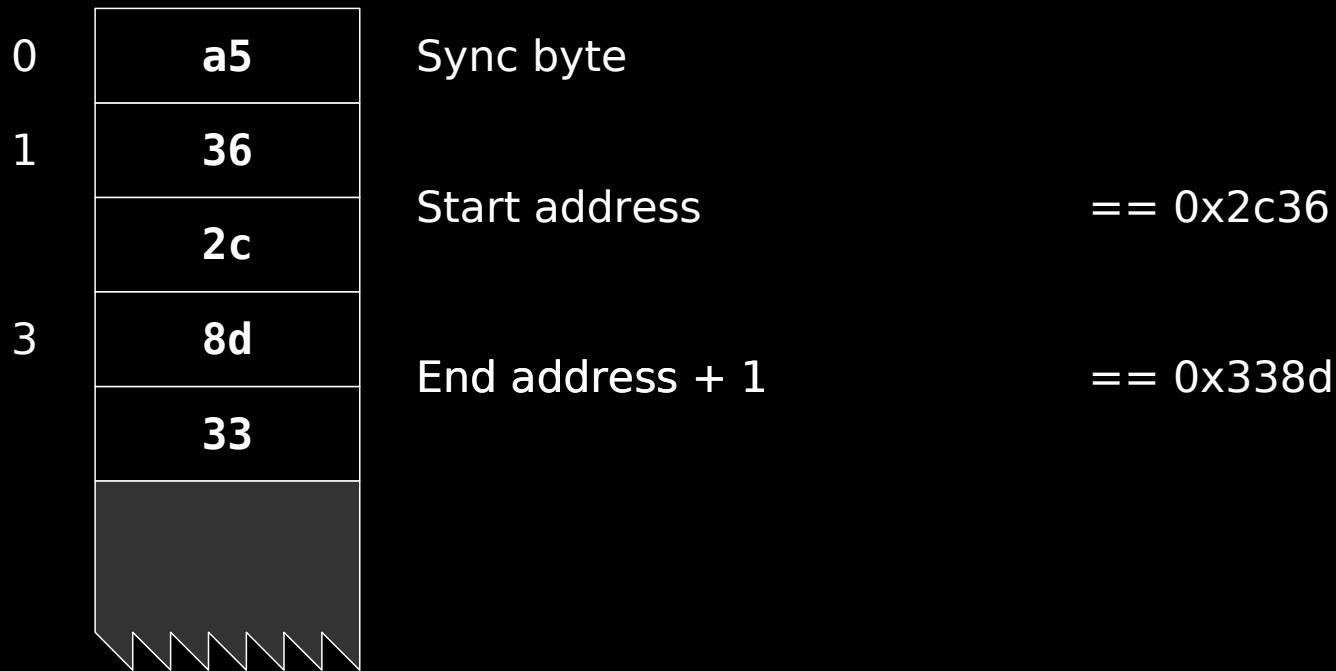




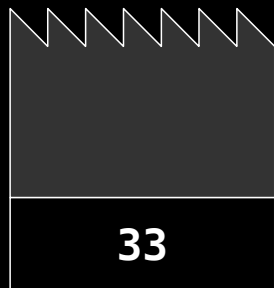
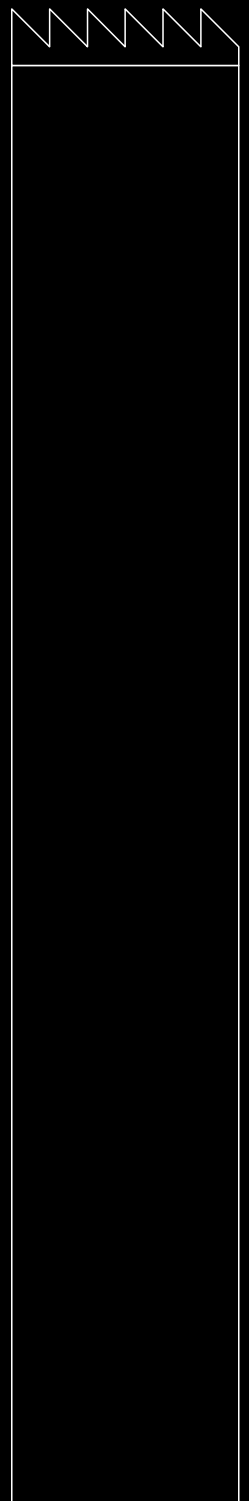




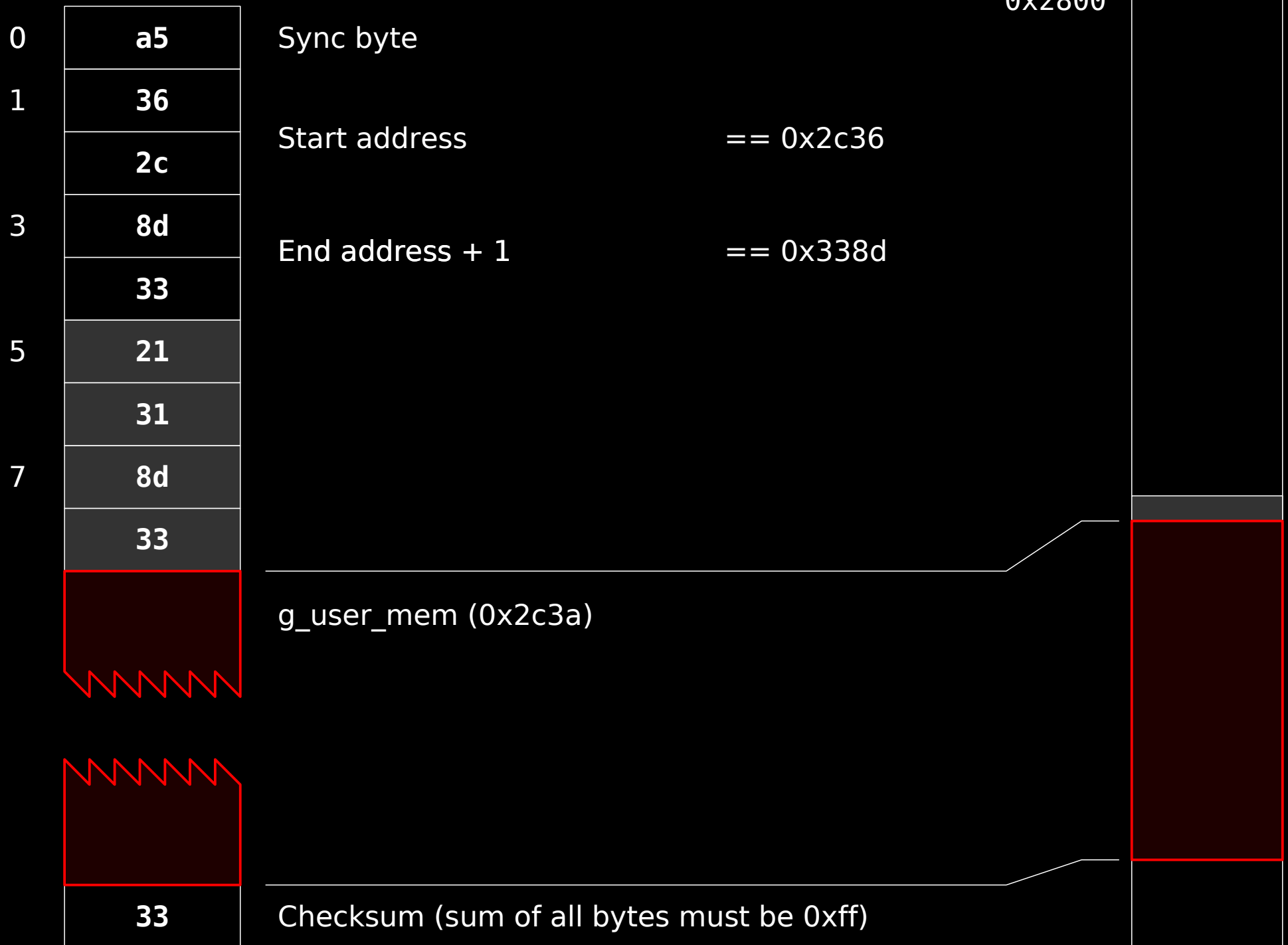




0x2800

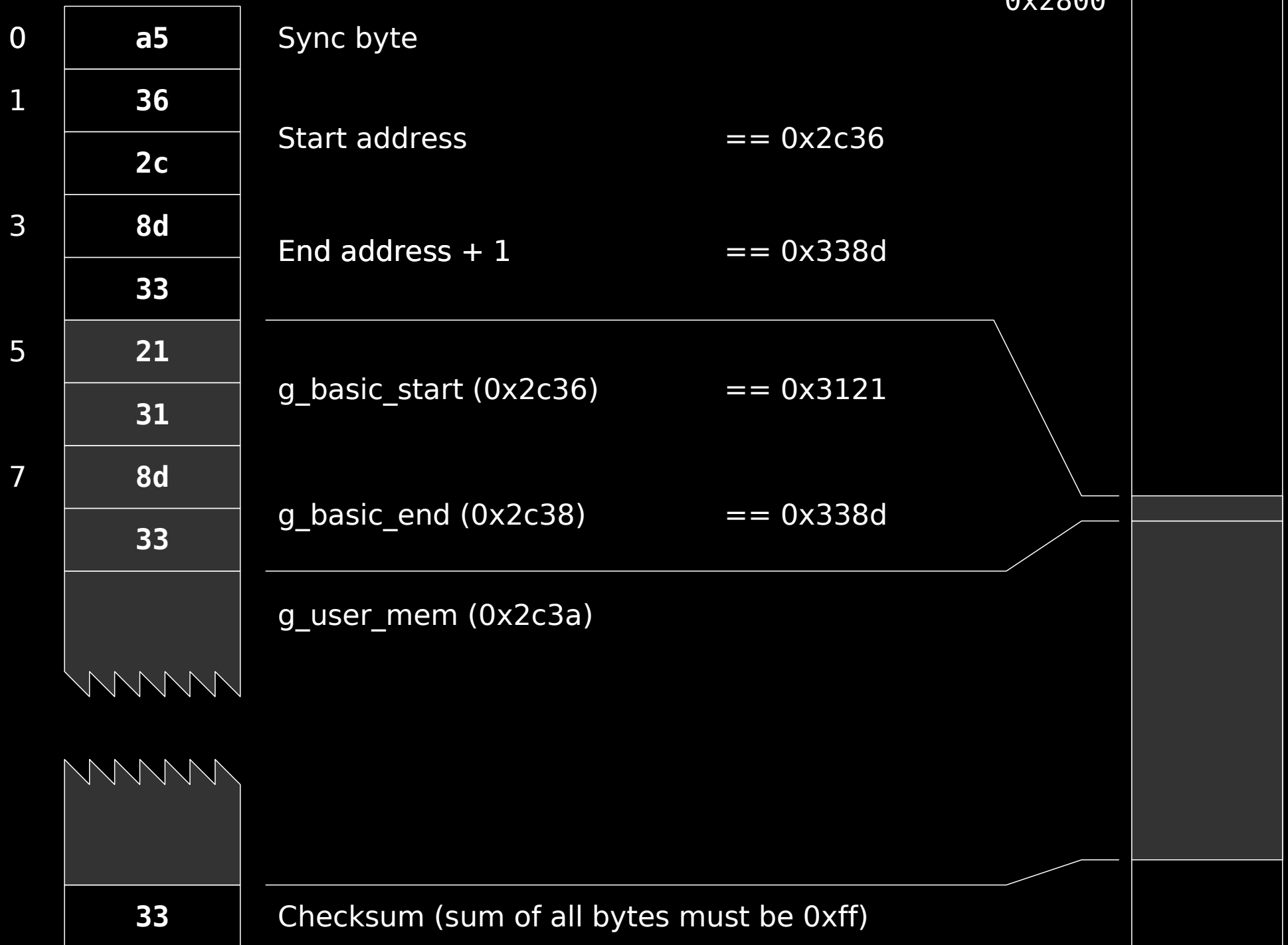


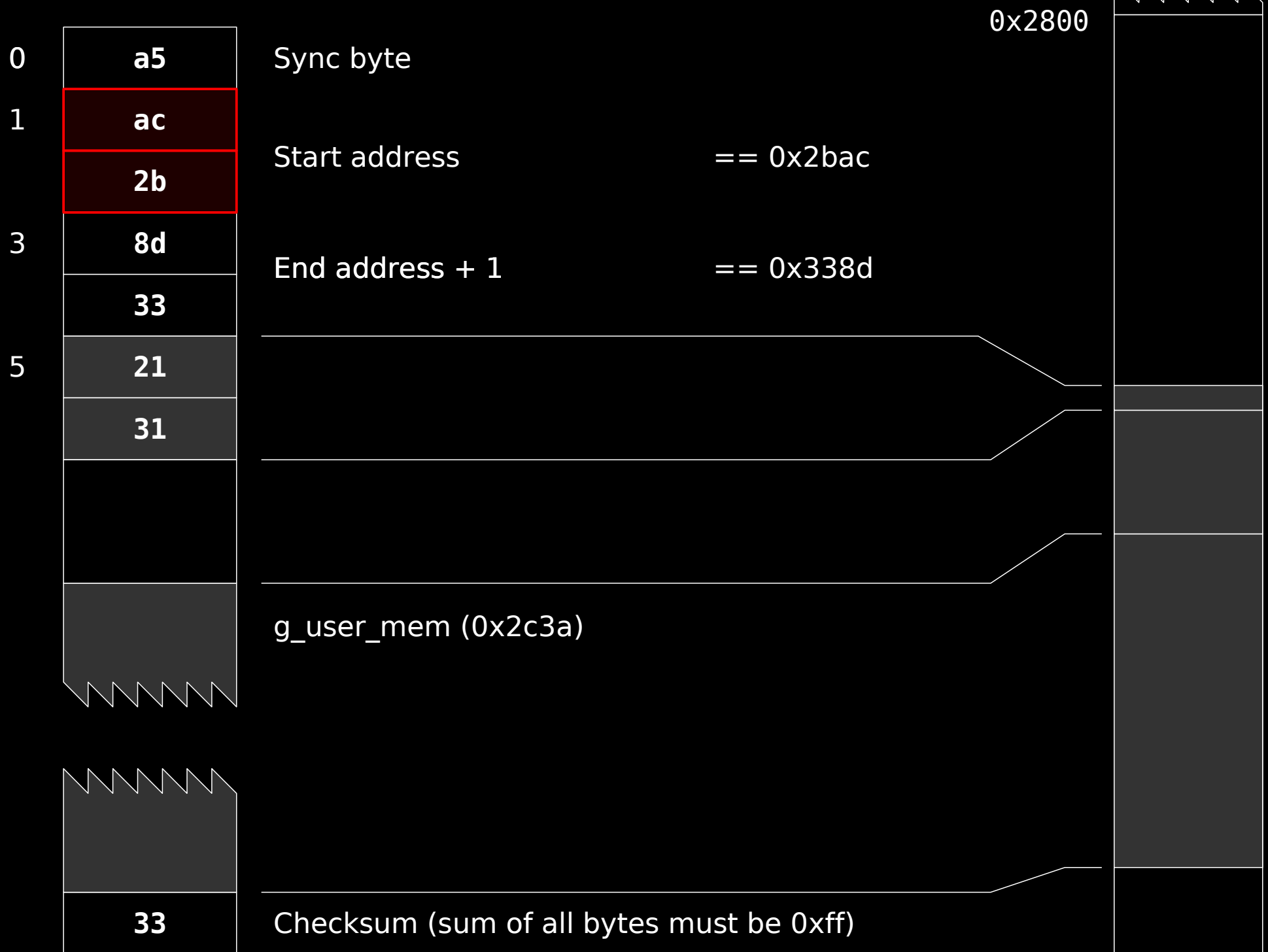
Checksum (sum of all bytes must be 0xff)

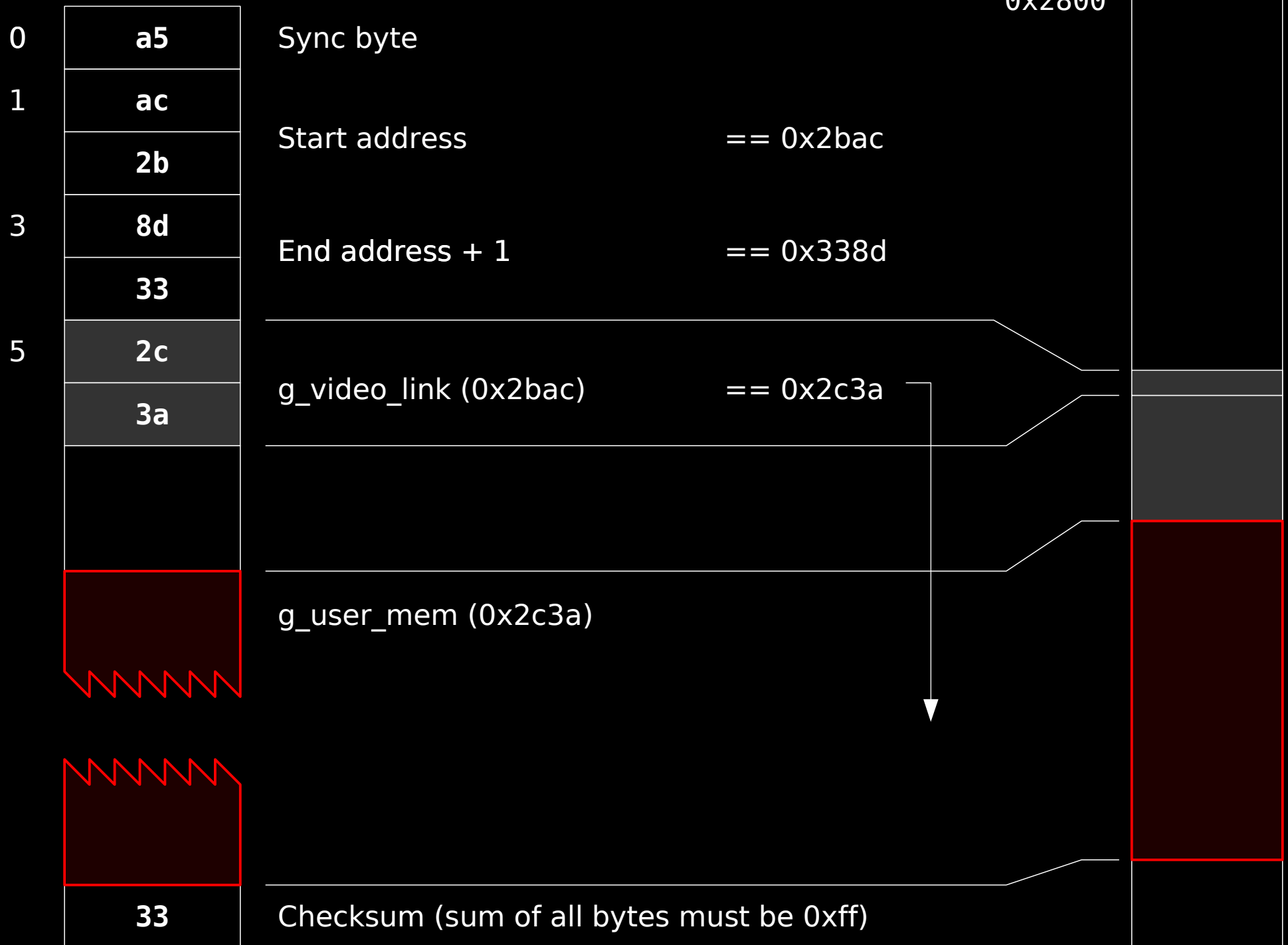


⇔ READY

>A=USR(&2C3A)_

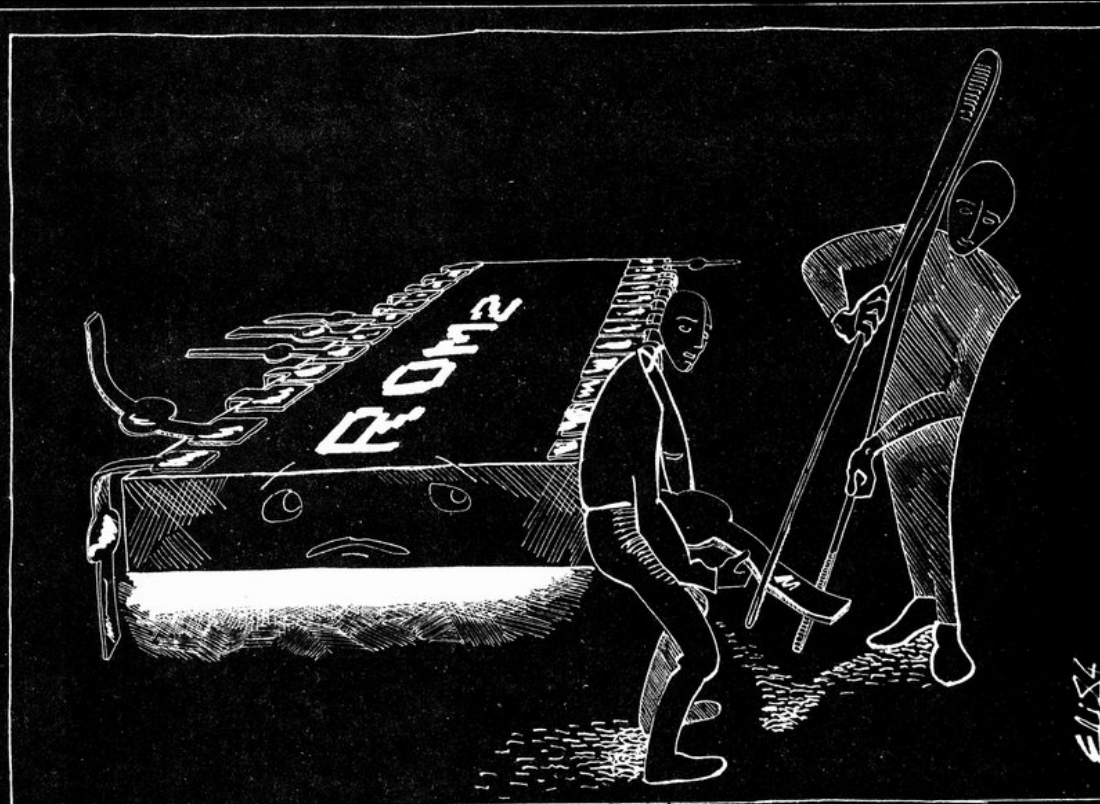






Upgrades

- Z80 assembler & development tools
- floating point functions
- BASIC extensions



37/ rom 2

računar
„galaksija“
2 ROM
Dejan Ristanović
Časopis „Galaksija“,
Beograd
Zavod za udžbenike
i nastavna sredstva, Beograd
Elektronika inženjering, Zemun

Upgrades

NAŠ TEST

GALAKSIJA PLUS

Na prvi pogled Galaksija Plus od obične Galaksije (ili Galaksije Minus, kako je Vojta Antić odmah nazvao) razlikuje se po veličini. To je, pored otvora za zvučnik, jedini znak da se ne nalazimo pred Galaksijom Minus. Sa zadnje strane računara nalazi se izlaz za TV (slika je na VHF području oko 36 kanala), petopinski kompozitni monitorski izlaz (tu je ujedno i izlaz za audio pojačalo), petopinski priključak za ispravljač i kasetofon, te jedan 44-polini i jedan 24-polini port za periferijsku opremu (na primer štampač). Galaksija Plus može se priključiti na bilo koji kasetofon koji ima ulaz za mikrofoni i izlaz za slušalice. Odmah se primjećuje nedostatak priključaka za palice za igre, koji ni ovdje, kao ni na Galaksiji Minus, iz nepoznatih razloga nisu stavljene. Iznutra, pored profesionalno urađene dvostrane štampane pločice (uz možda malo rasipanje prostora), prvo što se primjećuje jeste prazan prostor sa desne strane računara - ostavljen je za neki buduću interfejs za disk (baziran najvjerovatnije na tipu 1771).

Grafika i zvuk

Pored toga što se na ekranu prikazuje tekst u formu 16 redova sa po 32 znaka u redu, Galaksija Plus posjeduje dva grafička moda. U prvom modu niske rezolucije, Galaksija Plus može na ekranu da prikaže maksimalno 64x48 tačaka, i u njemu je potpuno kompatibilna sa Galaksijom Minus. U modu visoke rezolucije Galaksija Plus na ekranu prikazuje 256 tačaka po horizontali i 208 tačaka po vertikali (tj. i u tom slučaju za potrebe video memorije rezervišu se još 6656 bajta iznad RAMTOP-a. Bez obzira na mod u kojem radi, Galaksija Plus (kao i Galaksija Minus) daje crno-belu sliku (što se obično naziva slikom u dve boje), i može se slobodno reći da se to nikada neće promijeniti.

Slika na TV može ponekad (naravno! posle dužeg rada, na primer posle 20 do 30 sati) da postane jedva primetno nestabilna, te se zato preporučuje rad sa monitorom.

Zvučne mogućnosti Galaksije Plus dosta su dobre. Ton generator je baziran na tipu AY-3-9310 koji je registrarski orijentisan programabilni generator zvuka. Komunikacija sa ton generatorom vrši se preko I/O memorijske mape. Generator sadrži 16 registrara koje kontroliše mikroprocesor i programskom kontrolom moguće je postići 8 oktava na tri potpuno nezavisno kontrolisana kanala. Pored toga, u ovom tipu nalazi se i generator šuma (koji daje frekventno modulirane pseudoslučajne impulse pravougaonog oblika), generator envelope, miksera (koji omogućuju kombinaciju tonovskih generatorsa i generatora šuma za svaki kanal posebno) i kontrolu amplitude.

Vrlo je interesantno da je dizajner (teško da je to bila želja konstruktora) odlučio da potencijometar za podešavanje nivoa zvuka na ugrađenom zvučniku postavi na najnepristupačnije moguće mjesto - na donju stranu računara.

Tastatura

Tastatura je poluprofesionalnog tipa sa 57 tipki, ali i pored toga vrlo je ugodna za kucanje. Šteta da se niko nije setio da sada, kada takva mogućnost postoji, obezbjedi da pritisak na tipku proizvode i neki ton (naravno uz mogućnost isključivanja ove

Piše Nenad Balint

Pojava Galaksije Plus izazvala je veliko interesovanje čitalaca, vlasnika Galaksije Minus (kako staru dobru Galaksiju naziva njen konstruktor Vojta Antić), ali i onih koji je nemaju.

opcije). Pored standardnih slova, brojeva i specialnih znakova koji su već viđeni na Galaksiji Minus, pažnju privlače tipke sa našim slovima Š, Č, Ć, Ž. Ali, da bi se napravilo mesta za njih tipka sa znakom - je pomeren na gornji red, između tipki sa znakovima : i =. Dodavanje ovih tipki omogućuje to je vrlo jednostavnim trikom: naime, pritisak re-

cimo na S će javiti ROM-u 1 da su pritisnute tipke S i SHIFT (što se postiglo specijalnim prevezivanjem tastature). Naravno, opet je šteta što se konstruktor nije ranije setio da ovaj trik primeni i na tipke SHIFT i 0 i tako dobije novu tipku INSERT koja se koristi u ekranom editoru. Nova tipka bi se, na primer mogla staviti na mesto tipke STOP/ENTER, dok bi se STOP/ENTER pomorio u gornji red pored BRK (gde bi činio manje štete nego do sada). Inače, sve ovo je konstruktor Nenad Dunjić pomenuo u razgovoru sa autorom članka svega nekoliko dana pre zaključivanja lista kao sjajnu mogućnost, koja bi mogla da se možda i ostvari.

Novo u BASIC-u

Pored standardnog Galaksijinog, sporog, Basica iz ROM-a 1 i njegovih proširenja u ROM-u 2, Galaksija Plus dodaje je nekoliko novih naredbi. Prešli između dva grafička moda omogućeni su dve-



ma jednostavnim naredbama: TEXT (za nisku rezoluciju) i GRAPH (za visoku rezoluciju) koja ujedno i spušta RAMTOP za 6656 bajtova koje rezervišu za video memoriju ekrana visoke rezolucije. Naredbom TEXT se RAMTOP ne vraća na staro mesto. Mnogim programerima koji budu željeli da pšu sistemske programe sigurno će se svideti što se sve što se prikazuje na ekranu visoke rezolucije (osim naravno grafičke visoke rezolucije) prenosi i na ekran niske rezolucije. Zatim, vrlo lepo je rešeno i sa memoriziranjem memorije za ekran visoke rezolucije. Naime, jednostavnom promenom RAMTOP-a i sistemske promenljive u kojoj se čuva viši bajt adrese početka ekranske memorije omogućeno je brza izmena više slika na ekranu i to čak i iz Basic-a, pri čemu se u memoriji može nalaziti najviše 7 slika.

Naredbe za rad sa visokom rezolucijom su PLOT i DRAW za crtanje tačke i povlačenje linije i UNPLOT i UNDRAW za njihovo brisanje. Za sve četiri naredbe potrebna su samo dva parametra: x i y koordinata tačke koju treba nacrtati (izbrisati) ili do koje treba povući (izbrisati) liniju. Pri tome treba obratiti pažnju na to je DRAW apsolutan tj. daje mu se koordinate x i y koordinate tačke do koje treba povući liniju od prethodno nacrtane tačke, a ne relativno rastojanje po x i y koordinati od prethodno nacrtane tačke.

Logičan naslednik naredbe DRAW je naredba ELLIPSE koja, kao što joj i ime kaže, omogućava crtanje elipse ili, u specijalnom slučaju, jednakih poluosa, kruga. Ovoj naredbi potrebno je neobičajno mnogo parametara: pored koordinata jedne od žiža, veličina male i velike poluse i koraka sa kojim se crta, tu se nalaze i informacije da li je elipsa crna ili bela (tj. da li se briše ili crta) i podatak koliki je ugao između velike poluse i x koordinata. Jedini moguć komentar je: ovo je najkompleksnija naredba koju Galaksija Plus ima ili će imati.

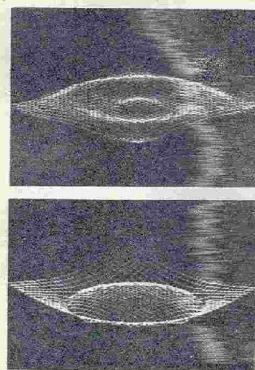
I na kraju naredbi za liniju grafičke jeste naredba FILL koja služi za ispunjavanje neke zatvorene konture, a kojoj su parametri samo x i y koordinate tačke od koje treba početi ispunjavanje. U trenutku kada se ovaj članak predavao u štampu ova rutina je bila puna bagova.

Jedina naredba koja podržava rad sa generatorom tona je SOUND. Simaksa naredbe je SOUND rs gde je r broj jednog od 16 registrara koliko ton generator ima i u koji želimo da upišemo sadržaj s. Ova naredba u sebi sadrži zapravo dve naredbe: OUT 0, r, OUT 1, s. Postojanje samo ove jedne naredbe čini rad sa generatorom tona vrlo nekonfornim (naravno u Basicu, u assembleru inače morate da koristite naredbu OUT). Biće pravdo čudo ako neko korišćenjem samo ove jedne naredbe bude uspeo da od generatora tona izvuče nešto više od običnih kratkih efekata pucaja, lasera, sirene, eksplozije i slično.

Rad sa kasetofonom uvedenim novih rutina QSAVE i QLOAD drastično se menja. Pored toga što je brzina povećana na 1200 boba (što predstavlja nekih 8 Kb u minuti) uvedeno je i pojavljivanje linija na ekranu. Takođe, predviđen je rad sa imenovanim datotekama (pri čemu se učitavanje prekida kada se učitava ime datoteke), a osim toga postoji i mogućnost snimanja programa koji će se po kasnijem učitavanju automatski startovati. Sve ovo sigurno vas posleđa na Spectrum i Commodore, zar ne?

Ekraniski editor

Novi editor koji je Galaksija Plus dobila predstavlja korak od 65 milja u odnosu na editor koji Galaksija Minus ima. Onih pola linije do nedostaje zbog takozvanog OVERTYPE moda u kojem ekraniski editor radi (tj. posle pozicioniranja kursora u okviru neke linije koju želite da editujete svaki



tekst koji budete kucali će se prepisati preko postojećeg sadržaja te linije). Autori ovog teksta je, ne jedinstven, u toku rada nedostajao INSERT mod u kojem radi editor Galaksije Minus. Uvođenje ovog tastera za INSERT ovaj nedostatak bi sigurno umanjilo, ali bi se tada sigurno javili oni kojima nedostaje SHIFT i strelica levo i desno za pozicioniranje kursora na početak odnosno kraj reda, zar ne?

Koliko je ovaj ekraniski editor moćan sigurno će primetiti svi koji pišu neki program (bilo na assembleru, bilo na Basicu) budu koristili naredbu za pretraživanje teksta (tj./). Ispravljanje pojedinih naredbi tada postaje prava pesma.

Neke nove mogućnosti

Postojanje sistemske promenljive u kojoj se nalazi broj linija koje se crtaju na ekranu pomena vrednost Benchmark testova bilo gde između normalnog (tj. sporog) i ubrzanog režima rada. Minimalan broj linija koje se crtaju na ekranu je 28 (negde oko dva reda), a maksimalan je 255 (tj. je naravno više od 16 redova koliko Galaksija podržava, pa se zato ostatak ne može kontrolisati: na dobrih programera je da dokažu suprotno). Sada ubrzan režim rada postaje mnogo interesantniji nego na Galaksiji Minus jer u dva reda teksta može da stane mnogo poruka.

Druga interesantna sistemska promenljiva jeste ona u kojoj se čuva adresa tabele znakova. S njom postaje vrlo jednostavno (odnosno manje komplikovano) definisanje novih karaktera, što otvara velike mogućnosti. Jedna od njih je pravljenje potpuno novog seta karaktera sa 50 znakova u redu i sa velikim i malim slovima (uostalom, kada to može da ima Spectrum koji ima isto 256 tačaka po horizontali, što ne bi i Galaksija Plus).

Bagovi

Galaksija Plus, kao ni mnoge bolj računari od nje, nije imuna na bagove. Srećom oni se odlažu, da tako kažemo, u bodu. Tako je, na primer, otklonjen jedan vrlo opasan bag izazvan nedostatkom prompta (upitnik, ako je u pitanju INPUT ili znak pri unošenju linije). Naime, ako se naredba PRINT završavala sa ; a odmah iza toga sledio INPUT onda su se dešavale dve stvari: ako je u pitanju bio numerički INPUT javljala se poruka WHAT?, a ako je u pitanju bio alfanumerički INPUT onda bi se sav tekst štampan naredbom PRINT stavio u alfanumeričku promenljivu u koju se vršilo učitavanje.

Otklonjen je i bag koji se javljao ako otvoreni znaci navoda ne bi bili zatvoreni. U ovom slučaju dolazilo je do efekta to što Galaksija Plus u trenutku otvaranja znaka navoda gubi ekraniski editor i na pritisak tipke sa strelicama pojavljuju se njihovi kontrolni kodovi koji služe za formatovanje ispisa. Ovi ređi razmaka ranije se prekidaio samo zatvaranjem znakova navoda, dok ga sada prekidaju i tipke ENTER i BRK.

Međutim, neki bagovi su ostali. Jedan od njih se javlja samo u GRAPH modu: ako koristite naredbe PLOT i DRAW nacrtate nešto u poslednje dve linije na ekranu, posle toga pokušate da listate program, primetićete da se one ne samo ne brišu već se u toku listanja pomeraju nagore zajedno sa tekstom. Ovo je jedan od najbezopasnijih bagova i otklanja se jednostavnim pritiskom na tipke SHIFT i DEL čime će biti izbrisani sadržaj celog ekrana.

Drugi bag koji je ostao mnogo je opasniji. Naime, ekraniski editor koji se nalazi u Galaksiji Plus za svoj rad koristi 16 sistemskih promenljivih u kojima čuva podatke o tome koliko ima znakova u pojedinom redu na ekranu. Ako je taj broj znakova 32 onda Galaksija Plus zaključuje da taj i sledeći red predstavlja celu liniju. E, sad ako vi u toku editiranja programa neko programski liniju produžite na 32 znaka ili više onda bi joj Galaksija Plus (ali samo u slučaju da se ispod nje na ekranu nalazi neki tekst) pridružila i sadržaj sledećeg reda (koji se naravno ne bi uništio, već bi i dalje postojao, ali dupliran, tj. onakav kakav je bio i u sklopu linije koju ste editovali). Sumnjamo da neko baš to može zaboraviti. Šteta što ovaj, vrlo neugodan, bag nije još uvek otklonjen.

Softverska podrška i hardverska proširenja

Osim jedne igre koja koristi generator tonova i nekih demonestracionih programa za visoku rezoluciju, praktično ne postoji nijedan program koji je posebno pisan za Galaksiju Plus. Srećom, potpuna kompatibilnost sa Galaksijom Minus omogućava korišćenje onog malo programa koji postoje za ovaj računar.

	BM1	BM2	BM3	BM4	BM5	BM6	BM7	BM8	prosek
Galaksija +	4,2	68	126	137	157	207	359	560	202
Galaksija + (ubrzana)	1,6	23	43	46	53	70	121	165	65
Spectrum	4,8	8,7	21	20	24	55	80	253	58
C64	1,4	10	19	20	21	32	51	116	34
BBC B	1	3	8,7	8,7	9	14	22	52	14,8
LOLA-8	5	14,3	44,5	49	52	77	102	295	80
ORAO	1,2	8	14	16	18	27	40	71	24

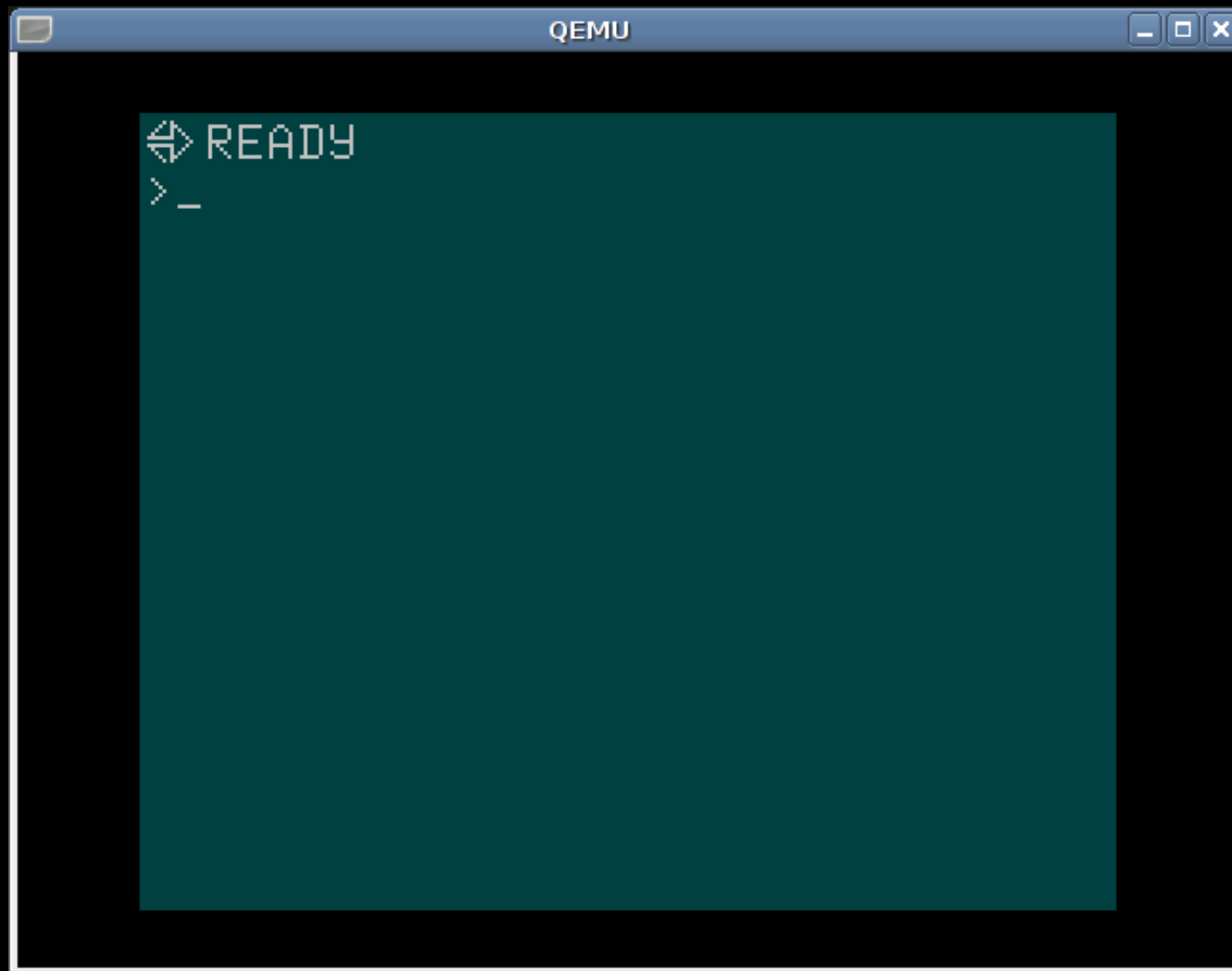
Nastavak na 21. str.

GalaxyWin - Emulator for Windows



<http://emulator.galaksija.org/>

Galaxy - Emulator for DOS



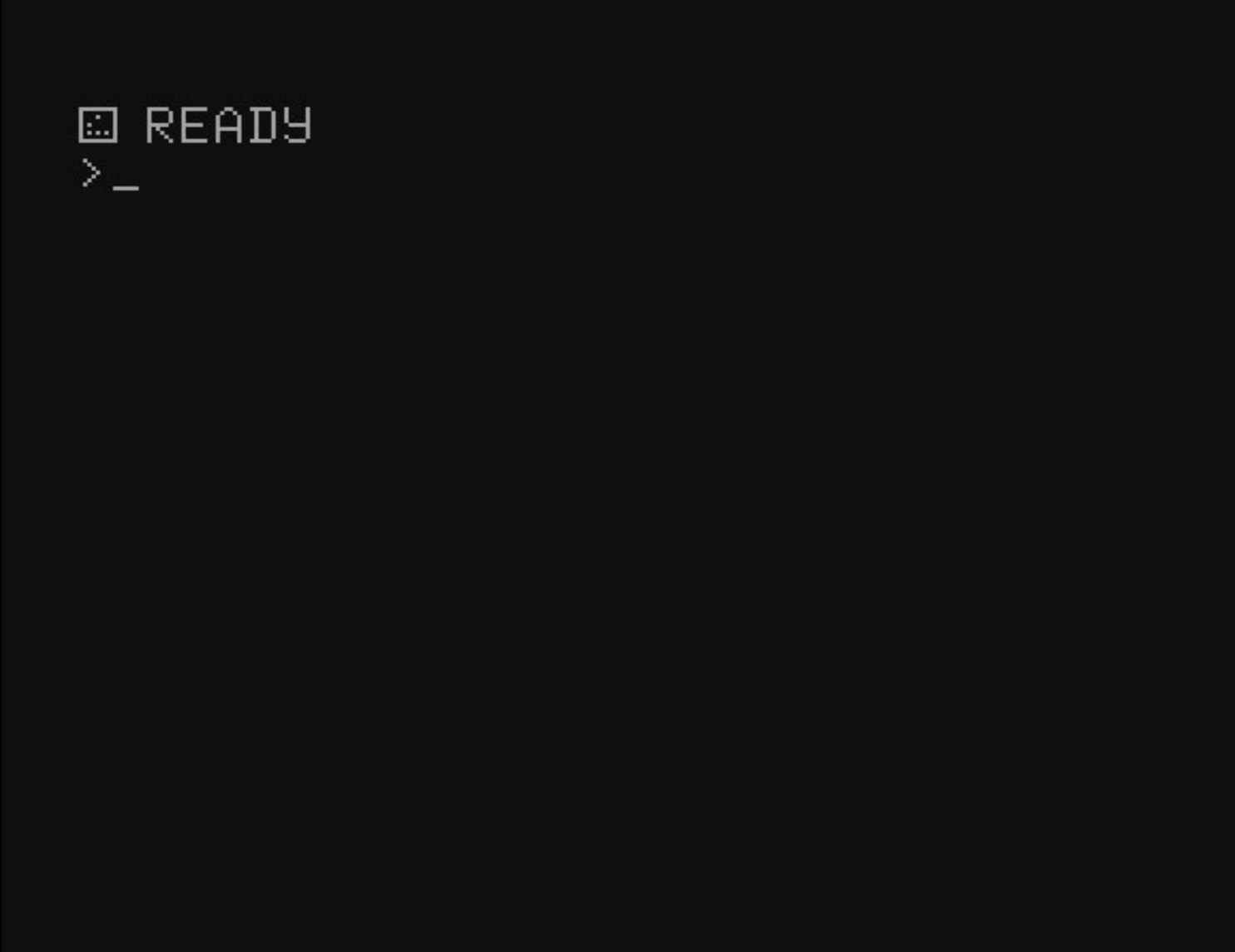
<http://solair.eunet.rs/~jovkovic/galaxy/>

Emulator for Sinclair ZX Spectrum 128k



<http://www.worldofspectrum.org/infoseekid.cgi?id=0017550>

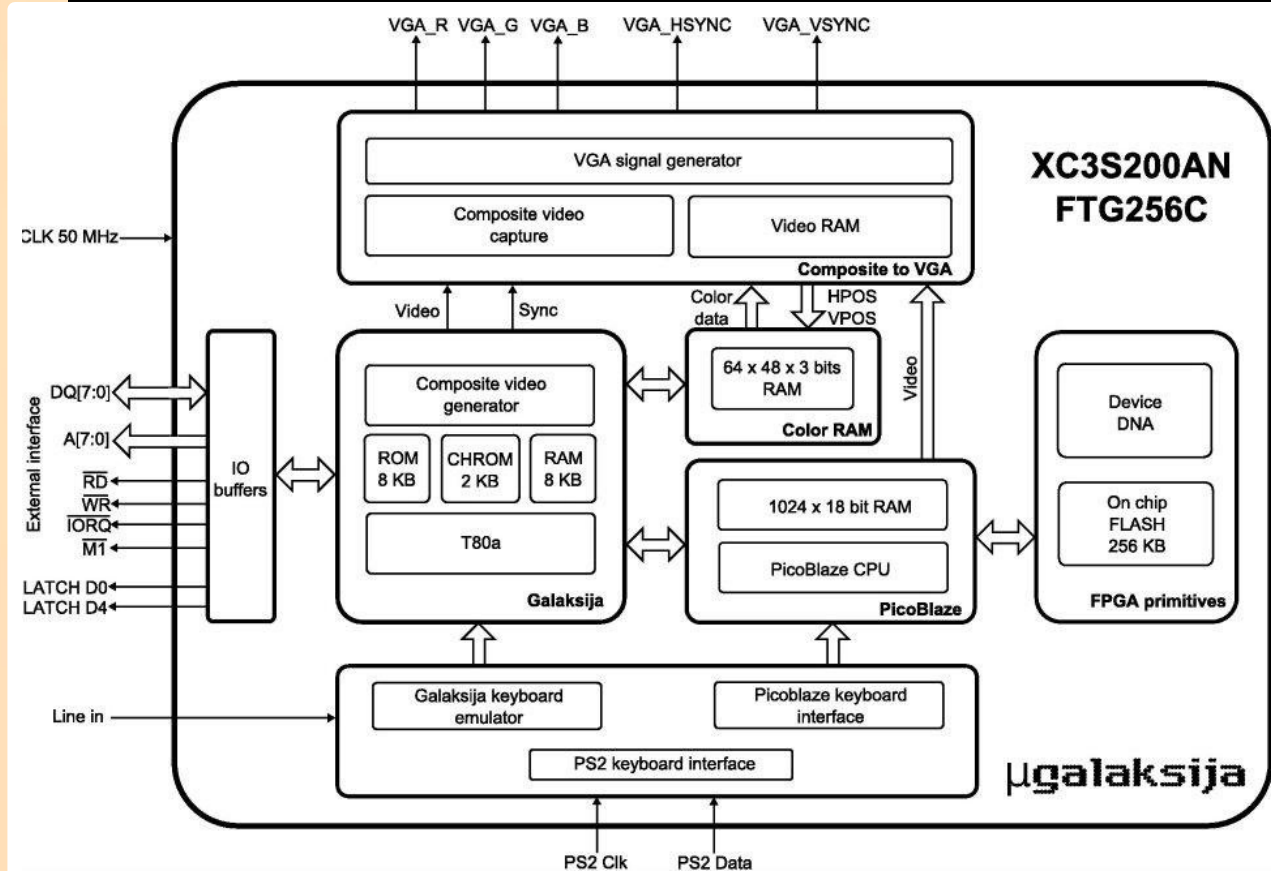
Emulator for SAM Coupé

A screenshot of a SAM Coupé emulator window. The window has a dark gray background. In the top-left corner, there is a small icon of a computer monitor. To the right of the icon, the word "READY" is displayed in a monospaced font. Below "READY", there is a prompt character ">" followed by an underscore "_".

```
READY  
>_
```

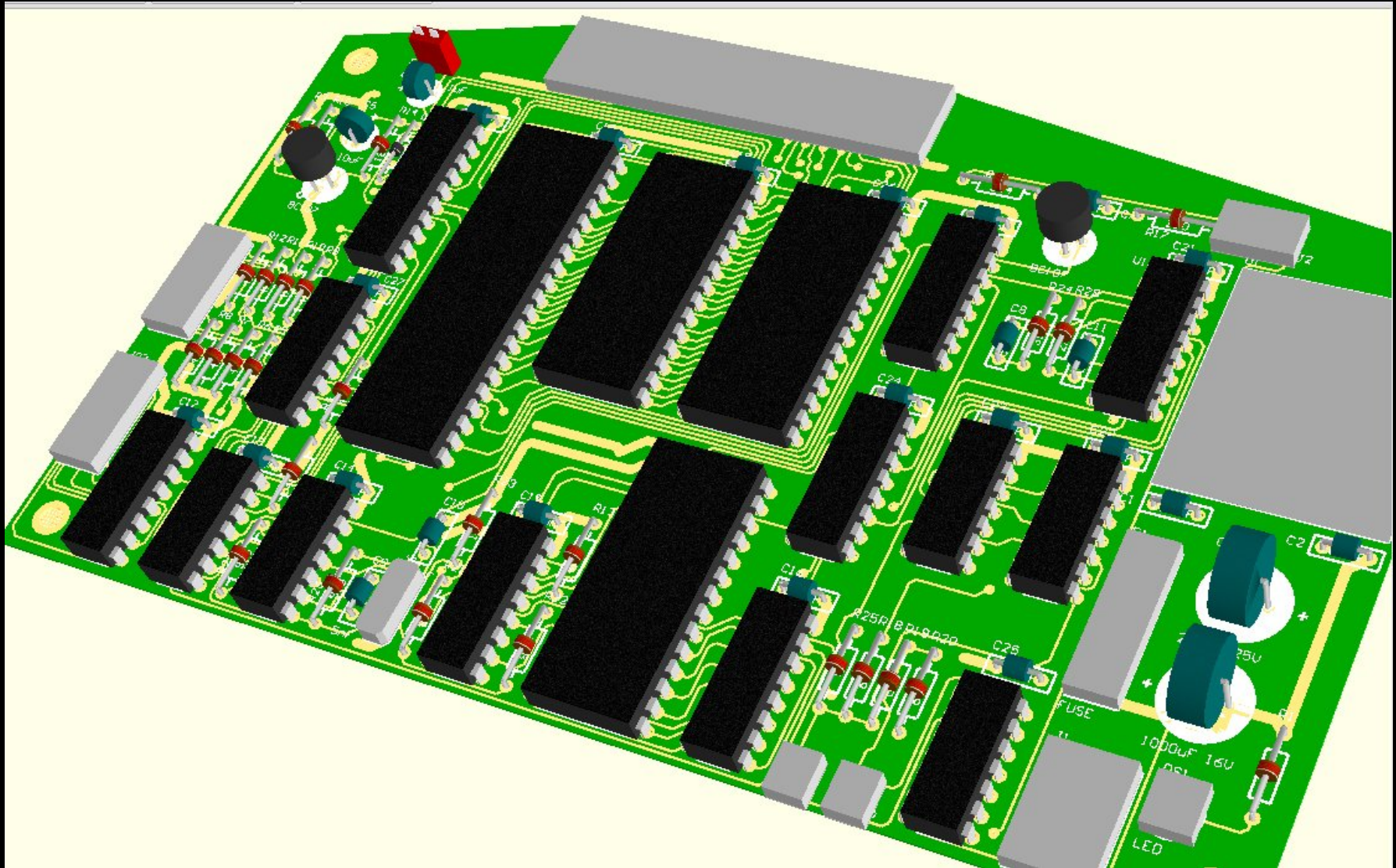
<http://simonowen.com/sam/galemu/>

μGalaksija - FPGA implementation



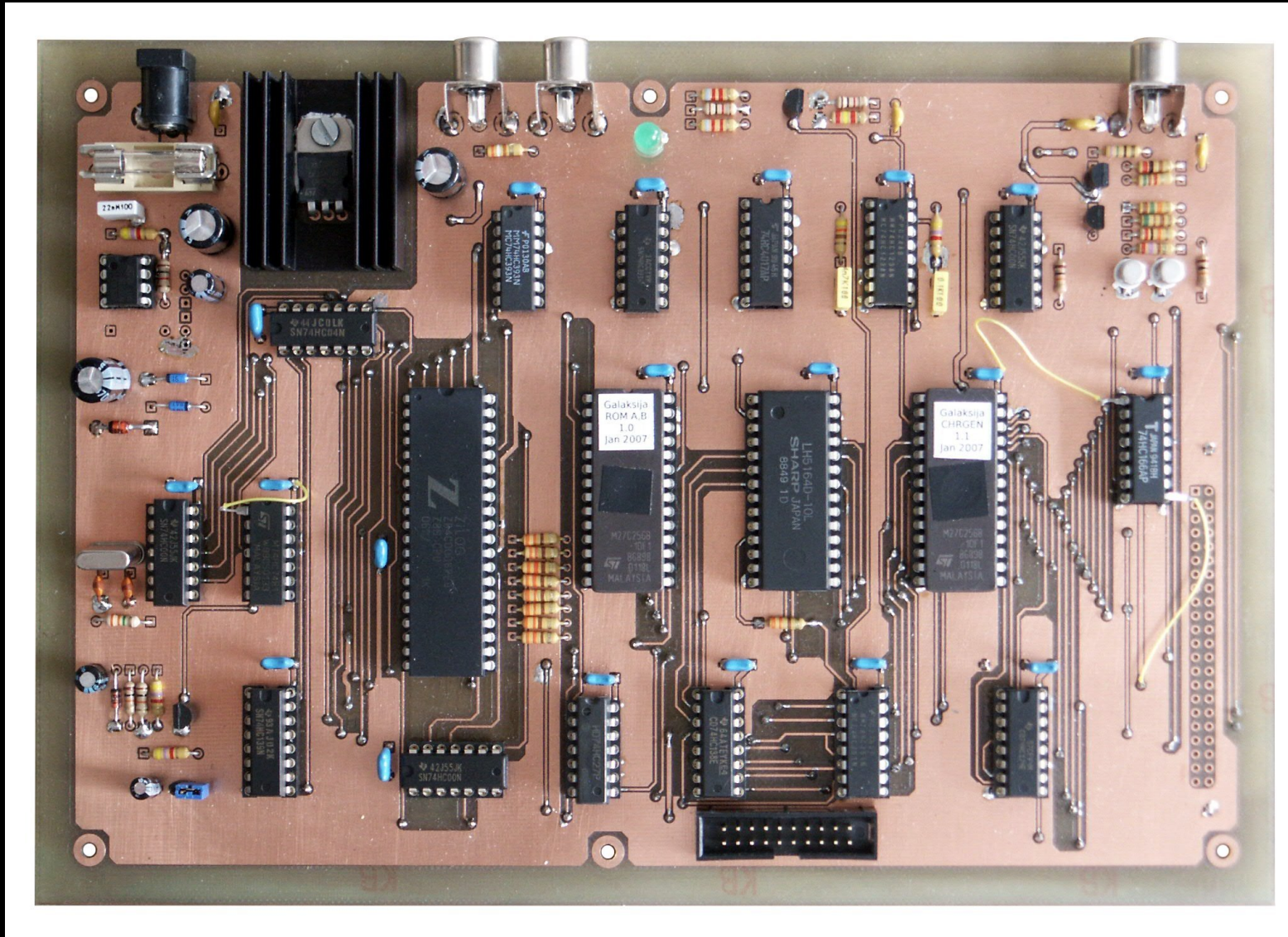
<http://galaksija.petnica.rs>

Galaksija - PLD implementation



<http://www.galaksija.org/pdf>

Galaksija - CMOS implementation



<http://www.tablix.org/~avian/blog/articles/galaksija/>

Development tools

- z80asm – Assembler
<http://www.nongnu.org/z80asm/>
- z80dasm – Disassembler
<http://www.tablix.org/~avian/blog/articles/z80dasm/>
- z88dk – C compiler
<http://www.z88dk.org/wiki/doku.php?id=platform:galaksija>
- Galaksija development tools
<http://www.tablix.org/~avian/blog/articles/galaksija-tools/>

Further information

- Home of the Z80 CPU
<http://z80.info>
- The incomplete ROM disassembly
<http://www.tablix.org/~avian/galaksija/rom/rom1.html>
- Collection of original software
http://retrospec.sgn.net/users/tomcat/yu/Galaksija_list.php
- Magazine scans
<http://emulator.galaksija.org/MagScans/index.html>
- Avian's Blog
<http://www.tablix.org/~avian/blog/search/?q=galaksija>
- Forum
<http://www.galaksija.org>

Questions?

tomaz.solc@tablix.org

@avian2

<http://www.tablix.org/~avian/blog>