1. Flash Content Arrangement

Address	Description
0x0000	Firmware length high byte
0x0001	Firmware length low byte
0x0002	Checksum high byte
0x0003	Checksum low byte
0x0004	Ram version high byte
0x0005	Ram version low byte
0x0006	VID high byte
0x0007	VID low byte
0x0008	PID high byte
0x0009	PID low byte
0x000A	Firmware Flag1
0x000B	Firmware Flag2
0x000C	Firmware Flag3
0x000D	Firmware Flag4
0x000E	Firmware Flag5
…	…
0x0100	Firmware start address

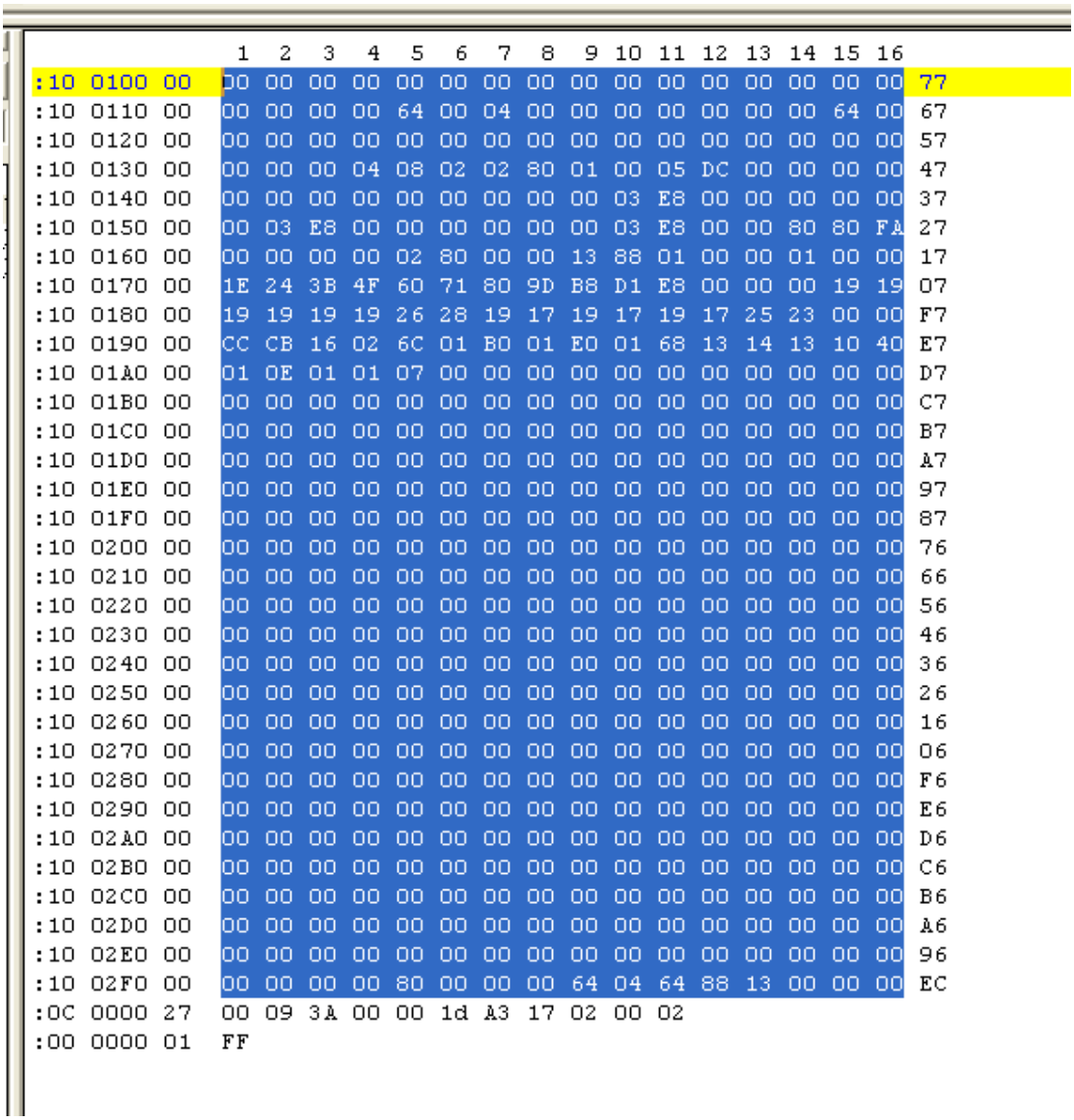
1. Checksum = Sum of (0x0100+…+Firmware length)
2. If checksum is error, firmware still running but it can be known by reading ISP register and firmware flag reset to default value 0x00.
3. After loading from flash, EPR_CSB pin will go high and flash will enter suspend mode.
4. VID and PID can modify by written it in EEPROM.

2. *UI.hex content

1. *UI.hex mapping to flash address 0x0000~0x02ff,

2. 0x0000~0x000f need mapping like “Flash Content Arrangement” item

3.Firmware length and firmware Checksum start from flash address 0x0100

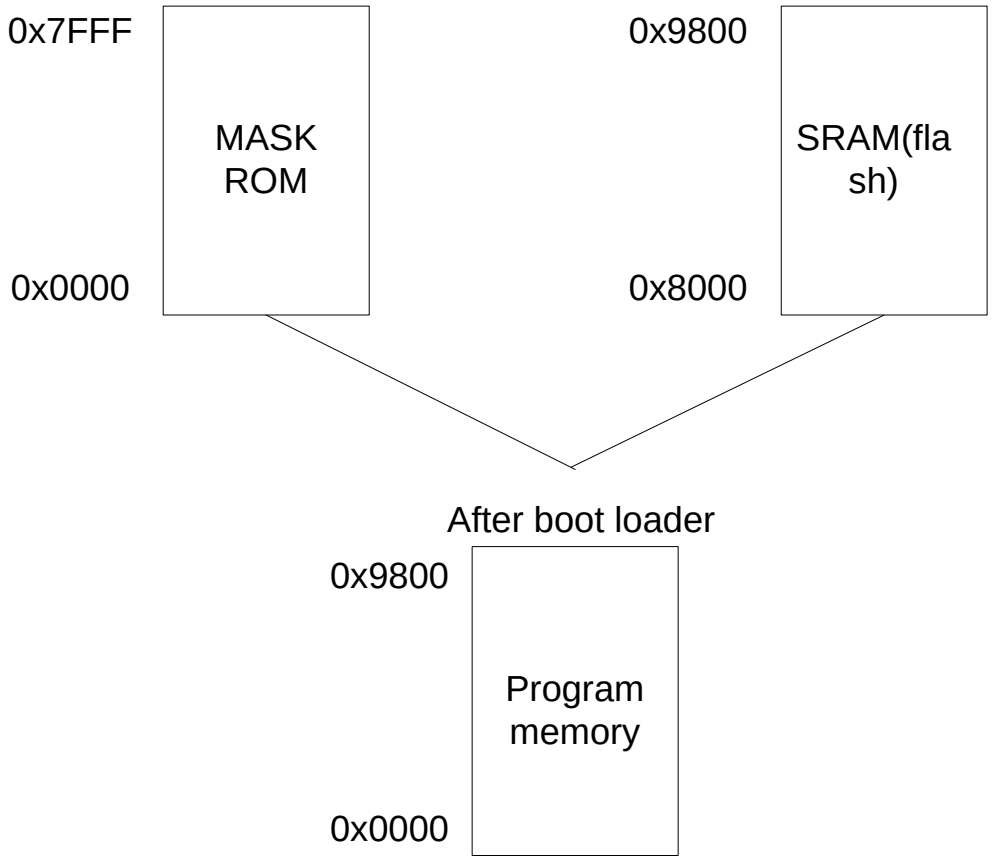


```
                  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
:10 0100 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  77
:10 0110 00   00 00 00 00 64 00 04 00 00 00 00 00 00 00 64 00  67
:10 0120 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  57
:10 0130 00   00 00 00 04 08 02 02 80 01 00 05 DC 00 00 00 00  47
:10 0140 00   00 00 00 00 00 00 00 00 00 03 E8 00 00 00 00 00  37
:10 0150 00   00 03 E8 00 00 00 00 00 00 03 E8 00 00 80 80 FA  27
:10 0160 00   00 00 00 00 02 80 00 00 13 88 01 00 00 01 00 00  17
:10 0170 00   1E 24 3B 4F 60 71 80 9D B8 D1 E8 00 00 00 19 19  07
:10 0180 00   19 19 19 19 26 28 19 17 19 17 19 17 25 23 00 00  F7
:10 0190 00   CC CB 16 02 6C 01 B0 01 E0 01 68 13 14 13 10 40  E7
:10 01A0 00   01 0E 01 01 07 00 00 00 00 00 00 00 00 00 00 00  D7
:10 01B0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  C7
:10 01C0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  B7
:10 01D0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  A7
:10 01E0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  97
:10 01F0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  87
:10 0200 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  76
:10 0210 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  66
:10 0220 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  56
:10 0230 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  46
:10 0240 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  36
:10 0250 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  26
:10 0260 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  16
:10 0270 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  06
:10 0280 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  F6
:10 0290 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  E6
:10 02A0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  D6
:10 02B0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  C6
:10 02C0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  B6
:10 02D0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  A6
:10 02E0 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  96
:10 02F0 00   00 00 00 00 80 00 00 00 64 04 64 88 13 00 00 00  EC
:0C 0000 27   00 09 3A 00 00 1d A3 17 02 00 02
:00 0000 01   FF
```

3. *FW.hex

start address *FW.hex 0x8200 content, mapping to flash address 0x0300

4.PAC7332 internal CPU Address Mapping



PCA7322 CPU Memory Address Diagram

1. There is an internal ROM using address 0x0000 to 0x7FFF and an internal RAM using address 0x8000 to 0x9800.
2. When power on, boot loader check EEPROM type and download EEPROM data to internal RAM.
3. After download, internal ROM and internal RAM could be merged. Firmware start running and ROM data could be access by address bus.
4. By USB vendor request, we can update EEPROM content.