

Microcontrollers

ApNote

AP1613

additional file
APXXXX01 . EXE available

Bootstrap Loader Versions

Any current step of the SAB 80C166, C167 and C165 except of the respective bond-out chips implements an on-chip bootstrap loader.

Hans Sulzer / Siemens MC

Question:

Which version of the on-chip bootstrap loader is implemented in which derivative and step of the 80C166, C167 and C165 microcontroller families?

Answer:

Any current step of the SAB 80C166, C167 and C165 except of the respective bond-out chips implements an on-chip bootstrap loader. For the 80C166/83C166, the bootstrap loader was introduced at first with the CB-step, for the 88C166 (Flash) with the ES1-BA step.

All current bootstrap loaders expect the same number of bytes to be loaded: 32 bytes. The acknowledge byte sent back by the bootstrap loader is meant as an additional identification mark for the membership of a particular microcontroller family. Please see the table below for more details.

The way how the bootstrap mode is entered in the 80C166 family (ALE pulled high and NMI# activated after a hardware reset) differs from that in the C167/C165 family (P0L.4 pulled low after a hardware reset). Note however, that NMI# has still to be activated in C167 steps less than or equal to step AC due to an hardware bug in these steps.

Chip	Step	Bootstrap Loader	Acknowledge Byte	Activated via
C161RI	any	32 bytes version	D5h (halfduplex support)	P0L.4
C161V	-	not implemented	-	-
C161K/C161O	any	32 bytes version	B5h (currently)	P0L.4
C163	-	not implemented	-	-
C163-16F	-	not implemented	-	-
C164CI	any	32 bytes version	D5h (halfduplex support)	P0L.4
C165	any	32 bytes version	B5h	P0L.4
80C166/83C166	≤ CA	not implemented	-	-
	≥ CB	32 bytes version	55h	ALE + NMI#
88C166	ES-BA	not implemented	-	-
	≥ ES1-BA	32 bytes version	55h	ALE + NMI#
C167	ES-AC AC	32 bytes version	A5h	P0L.4 (+ NMI# due to a bug)
	≥ BA	32 bytes version	C5h (halfduplex support)	P0L.4
C167XX	any	32 bytes version	C5h (halfduplex support)	P0L.4