

Programowanie niskopoziomowe – assembler 8086.

Zad. 2 Programowanie przerwania czasowego - programy rezydentne.

Zaprojektować i uruchomić dwa programy w assemblerze 80x86:

1. Program rezydentny, który w przerwaniu 1CH BIOS, po odblokowaniu przez inny program (program nr 2 – parametr $N \neq 0$), N razy [byte] zapisuje w odstępach około 3 sekundowych dużą literę od adresu ADR [DWORD] (bufor obrazu w trybie tekstowym). Po uruchomieniu program sprawdza czy procedura rezydentna jest już zainstalowana w pamięci i zwraca informację o jej zainstalowaniu lub braku instalacji. Jeżeli program nie wykryje procedury automatycznie instaluje ją w pamięci komputera. Program używa parametru zewnętrznego „u” do usuwania procedury rezydentnej z pamięci komputera.
2. Program wprowadzający wielokrotnie dane do programu rezydentnego.
 - Formaty wprowadzania danych:
 - ADR - kod szesnastkowy [xxxx:xxxx] Zakres zmiennej ADR do 1MB.
 - B - znak 'A'..'Z' [x]
 - N - kod dziesiętny [xxx] <ENTER>
 - Tylko ostatnia dana jest potwierdzana klawiszem ENTER.
 - Program pozwala wprowadzić następne dane dopiero po wykonaniu poprzedniego zadania przez procedurę rezydentną ($N=0$), działanie procedury rezydentnej sygnalizowane jest komunikatem „Czekam zadanie w toku !”
 - Program działa w pętli tzn. po wprowadzeniu danych oraz zakończeniu zadania przez procedurę rezydentną pyta o kontynuację działania (ponowne wprowadzenie danych i uruchomienie procedury rezydentnej) Tak/Nie ? Program czeka na wciśnięcie klawiszy T lub N.