```
Struct CPUX
  mycpu (void) mangus montred to 114
  in inmapicia, is por our bonnes or
 if (readeflags() & FL-IF)
      panic (" my cpu called with interrupts
       enabled (n); modernot all bright
    apicid = lapicid();
     tor (1=0; ic nepu; ++1)
Engle (Copus (i). a picid = = apicid)
no prinner return & cpus (i))
   panic ("unknown apicid In");
         disables interuptioned and amoral
       End in processos many coluction
  struct procx
  myproc (void)
  & struct Cpu*(;
    struct proc * P)
    push di ();
   c= mycp4 ();
    P=c >proc;
    pop di();
   return p;
 3
   In many places, the processor must identify
```

8.

In many places, the processor must identify which process it is running, for example when a processor serves a field system call, the processor must determine which process is invoking field xus follows the standard plan that relies on a tiny

bit of hardware support

The kernel uses my cpu to find, its apicid, which in turn calls the function lapicid, The function lapicid returns the hardware identifier for this processer. my cpu identifier to find its struct cpu.

The kernel uses my process that is running on proc for the process that is running on current processor. The kinction my proc disables interuppt and mobber my upu to kind its processor state, which contains a field proc.

struct (pu x (

struct (poc x r)

puchelic);

pochelic);

populicly;

or crestor

The terms of and isolation of a second of the control of the contr