button

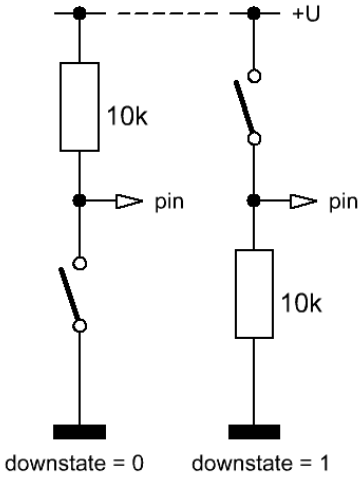
Platí pro všechny typy PICAXE

Syntaxe:

BUTTON pin, downstate, delay, rate, variable, targetstate, address

pin – je proměnná nebo konstanta, určující, který pin mikrokontroléru bude použit pro připojení spínače.

Downstate – je konstanta nebo proměnná datového typu bit (0 nebo 1), určující, jaký bude mít vstup logický stav při sepnutí spínače.



Je-li při sepnutí spínače na vstupu log.0 (0 V), (např. 10k pull-up rezistor s přepínačem připojeným na 0 V) zadejte 0.

Je-li po sepnutí spínače na vstupu log.1, (+U), (např. 10k pull-down rezistor se spínačem připojeným na +U) zadejte 1.

Delay – je proměnná nebo konstanta (1 až 254, 0 nebo 255), určující prodlevu před prvním automatickým opakováním sepnutí spínače. Hodnota delay 255 zakáže opakování, hodnota 0 zakáže opakování i prodlevu.

Delay je počítadlem, udávajícím počet průchodů testovací smyčkou před spuštěním funkce autorepeat. Jestliže je hodnota delay v rozsahu 1 až 254, bude uložena do pomocné proměnné bytevariable a dokud je spínač sepnut, po každém průchodu smyčkou se její hodnota zmenší o 1. K odskoku na návěští adress dojde v okamžiku, kdy hodnota bytevariable nabyde stavu 0. Počítadlo tedy nastavuje počáteční zpoždění před zahájením autorepeatu.

Pokud chcete pomocí zpožďovací smyčky jen ošetřit mechanické zákmity spínače a pak provést jednoduchý odskok na návěští bez následného automatického opakování, použijte hodnotu delay 255, která autorepeat zakazuje.

Hodnota delay 0 zakáže jak zpožďovací smyčku, tak funkci autorepeat a příkaz button bude fungovat jako jednoduchý příkaz „if PIN = targetstate then…".

Rate – je proměnná datového typu byte nebo konstanta (0 až 255) která určuje rychlost následného automatického opakování stisků tlačítka.

Hodnota rate určuje rychlost běhu testovací smyčky a tím určuje rychlost autorepeatu. Jakmile skončí testovací smyčka, bude hodnota rate uložena do pomocné proměnné bytevariable, a dokud je spínač sepnut, při každém průchodu smyčkou se hodnota rate sníží o 1. K odskoku na návěští adress dojde opakovaně vždy, když počítadlo dosáhne stavu 0. Tímto způsobem se nastavuje zpoždění mezi jednotlivými cykly autorepeatu.

Variable – je pomocná proměnná datového typu byte, která je používána pro čítač průchodů testovací smyčkou a smyčkou autorepeatu. Hodnotu proměnné je nutno v programu nastavit na 0 ještě před prvním použitím příkazu button (ještě před začátkem smyčky, uvnitř které se příkaz button používá).

Targetstate – je proměnná datového typu bit nebo konstanta (0 nebo 1), která určuje, při kterém stavu (0 = není stisknuto, 1 = stisknuto) spínače dojde v programu k odskoku na adresu adress. Tato hodnota může být použita pro invertování funkce spínače, k odskoku na danou adresu tedy může dojít, pokud bude spínač v okamžiku testu sepnut (targetstate = 1), nebo pokud bude naopak rozpojen (targetstate = 0).

Address – je návěští, na které program odskočí.

Popis:

Detekuje a reaguje na stisk tlačítka. Nastavitelným zpožděním eliminuje zákmity u mechanických spínačů. Umožňuje nastavení opakování klávesy jako u PC klávesnice.

Umožňuje větvení programu v závislosti na stavu spínače. Po sepnutí nebo rozepnutí spínače odskočí na definované návěští.

Pro informaci:

U mechanických spínačů s kovovými kontakty dochází díky pružnosti materiálu při sepnutí k několikanásobnému odskoku kontaktu a tím k jeho vícenásobnému sepnutí a rozpojení. Kontakt potřebuje určitou dobu (v řádech desítek ms) na uklidnění. To může způsobit potíže, protože mikrokontrolér tento stav nejspíše vyhodnotí jako několikanásobné sepnutí kontaktu.

Jedním z jednoduchých způsobů ošetření tohoto stavu je použít v programu krátkou pauzu (např. pause 10) a pak stav spínače znovu otestovat, což poskytne kontaktům spínače čas na uklidnění.

K ošetření tohoto problému nabízí PICAXE příkaz button. Po spuštění příkazu mikrokontrolér zkontroluje, zda je splněna podmínka downstate. Pokud je to pravda, proběhne čekací smyčka, která zamezí chybám, způsobeným odskoky kontaktu, a pokud je proměnná targetstate rovna 1, program přejde na návěští adress. Pokud je targetstate rovna 0, program pokračuje na dalším řádku.

Autorepeat pracuje stejně jako u počítačové klávesnice, pokud tedy stiskneme a držíme klávesu, odešle se jeden znak, následuje pauza a pak se začnou stejné znaky odesílat rychle za sebou.

Aby bylo možno využít všech popsaných možností příkazu button, musí být použit uvnitř smyčky. Pokud není umístěn ve smyčce, ale přímo v programu, pak příkaz jen jedenkrát zkontroluje stav spínače a rozhodne, zda dojde k odskoku na dané návěští, nebo zda bude program pokračovat dále na dalším řádku.

Pokud je příkaz button použit uvnitř smyčky, spouští se opakovaně a stav proměnné downstate je kontrolován opakovaně. Pokud stav stále platí, je do pomocné proměnné bytevariable uložena hodnota delay. S každým dalším průchodem smyčkou – pokud podmínka stále platí – se hodnota bytevariable snižuje o 1, dokud nedosáhne hodnoty 0. V tomto okamžiku, pokud se proměnná targetstate rovná jedné, dojde k odskoku na návěští adress. Do pomocné proměnné bytevariable se pak uloží hodnota rate a celý proces se opakuje, na konci každého průchodu smyčkou se proměnná bytevariable snižuje o jednu, dokud nedosáhne hodnoty 0, a pokud se proměnná targetstate stále rovná jedné, dojde k dalšímu odskoku na návěští adress.

Související příkazy:

- inputtype

Příklad:

Při stisku tlačítka na C.0, se rozsvítí LED na výstupu B.7 a pošle se zpráva na PC.

```
init:
b2 = 0                              ; vynulování pomocné proměnné

; vstup na C.0, stisk +V, když je tlačítko stisknuto skok na „pushed“

myloop:
button C.0,1,200,100,b2,1,pushed    ; skok, když je C.0 = 1
low B.7                             ; vypnutí LED
pause 10                            ; čekej
goto myloop                         ; opakování

pushed:
high B.7                            ; zapnutí LED
sertxd („PUSH“)                     ; odeslání zprávy
goto myloop                         ; skok na návěští „myloop“
```