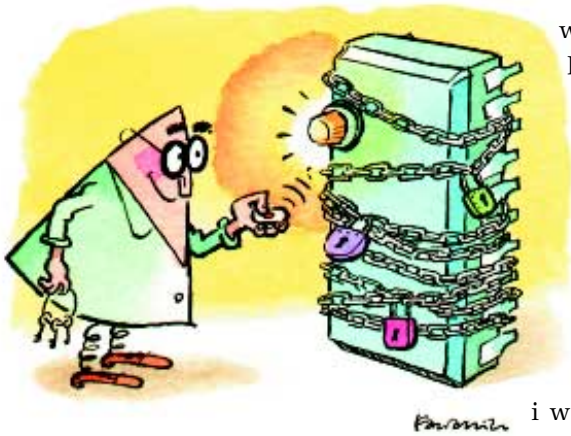


Bity blokujące i bezpieczniki konfiguracji w mikrokontrolerach AVR



Przemysław

Wprowadzenie

Nowoczesne mikrokontrolery oferują wiele opcji, które usta-

wiane są w zależności od potrzeb konkretnej aplikacji. Umożliwia to producentom wytwarzanie jednego mikrokontrolera o nazwie np. ATmega103 mającego możliwość pracy zarówno z generatorem zegarowym w postaci zewnętrznego elementu RC ustalającego częstotliwość jego pracy, jak i wewnętrznych, nie wykluczając również możliwości zastosowania oscylatora kwarcowego. Jest to sytuacja diametralnie inna od tej spotykanej w przeszłości, gdy dla przykładu niektóre mikrokontrolery wyposażone były w układ *watchdog* ustawiany albo programowo, albo działający bez względu na to, czy był potrzebny, czy też nie i wręcz przeszkadzał w poprawnym funkcjonowaniu aplikacji. Czasami takie mikrokontrolery różniły się tylko obecnością lub brakiem pewnej literki w nazwie wyrobu i nie mając świadomości tego, co się kupuje, można było zapłacić za rzecz, której zupełnie

Jaka jest różnica pomiędzy tzw. lock bit (bit zamykający, blokujący dostęp) a fuse (bezpiecznikiem)? Jakie są konsekwencje ich ustawiania czy też kasowania? Czym różnią się między sobą i czy ich ustawienia są odwracalne? W artykule postaram się odpowiedzieć na takie pytania. Myślę, że ta wiedza bardzo przyda się zwłaszcza stawiającym pierwsze kroki w dziedzinie mikrokontrolerów i posłuży lepszemu zrozumieniu podstawowych zasad korzystania z dostępnych zabezpieczeń.

nie dawało się użyć. Możliwość konfiguracji mikrokontrolera jest więc wygodna zarówno dla producenta, jak i dla konstruktorów. W niektórych modelach mikrokontrolerów zasadnicza różnica pomiędzy *lock bit* a *fuse* często

Tab. 1. Możliwe nastawy bitów blokujących oraz konsekwencje ich ustawienia

Tryb (Mode)	Stan bitów blokujących		Rodzaj włączonego zabezpieczenia
	LB1	LB2	
1	1	1	Niezaprogramowane, zabezpieczenie wyłączone
2	0	1	Zapis programu zabroniony, odczyt jest dozwolony
3	0	0	Zapis i odczyt programu są zabronione

Tab. 2. Dostępność bezpieczników w trybach programowania

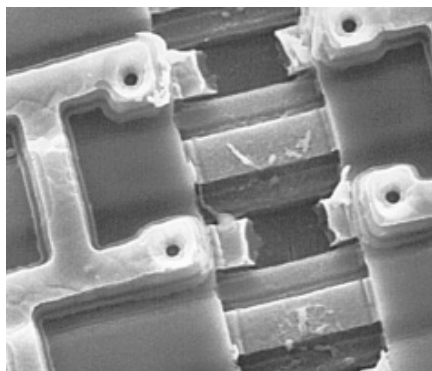
Mikrokontroler	RCEN	SPIEN	RSTDISBL	FSTR	BODEN	BODLEVEL	CKSEL n..0	EESAVE	SUT n..0	BOOTRST	BOOTSZ n..0	INTCAP
AT90S1200	R	R	-	-	-	-	-	-	-	-	-	-
AT90S2313	-	R	-	R	-	-	-	-	-	-	-	-
AT90S/LS2323	-	R	-	R/S	-	-	-	-	-	-	-	-
AT90S/LS2343	-	R	-	-	-	-	-	-	-	-	-	-
AT90S/LS2333	R/S	R	-	-	R/S	R/S	R/S	-	-	-	-	-
AT90S/LS4433	-	R	-	-	R/S	R/S	R/S	-	-	-	-	-
AT90S8515	-	R	-	R	-	-	-	-	-	-	-	-
AT90S/LS8535	-	R	-	R/S	-	-	-	-	-	-	-	-
ATtiny11	-	-	H/S	H/S	-	-	-	-	-	-	-	-
ATtiny12	-	H/S ¹	H/S ²	-	H/S	H/S	H/S	-	-	-	-	-
ATtiny15	-	H/S ¹	H/S ²	-	H/S	H/S	H/S	-	-	-	-	-
ATtiny28	-	-	-	-	-	-	-	-	-	-	-	R
ATmega103	-	R	-	-	-	-	-	R/S	R/S	-	-	-
ATmega161	-	R	-	-	P/S	P/S	P/S	-	-	R/S	-	-
ATmega163	-	R	-	-	P/S	P/S	P/S	-	-	R/S	R/S	-

R - programator równoległy, S - programator szeregowy (również programowanie *in-system*),

H - programowanie w trybie *High Voltage Serial Programming* (HVSP)

¹ Bezpiecznik jest dostępny w trybie programowania przez SPI, jednak jego wyzerowanie spowoduje, że dostęp w trybie programowania przez SPI nie będzie możliwy.

² Bezpiecznik jest dostępny w trybie programowania przez SPI, jednak jego ustawienie spowoduje, że dostęp w trybie programowania przez SPI nie będzie możliwy.



Fot. 1

polega na zupełnie odmiennym wykonaniu. Na przykład bezpieczniki (*fuse*) są wykonywane jako mikropołączenia w strukturze układu scalonego (fot. 1) albo też do struktury układu scalonego i fizycznie uszkodzane przy pomocy prądu elektrycznego w czasie programowania. Takie przepalone połączenie już nie daje się odtworzyć - układ nie nadaje się do ponownej zmiany danej nastawy czy też zmiany zawartości pamięci programu.

Inaczej jest z bitami blokującymi (*lock bits*). Ich ustawianie czy też kasowanie przypomina włączanie i wyłączanie opcji w programie - aplikacji. Wykonane są one najczęściej w technologii EEPROM, możliwa jest więc modyfikacja ich stanu. A jak to jest w przypadku mikrokontrolerów z rodziny AVR?

Mikrokontrolery AVR

Każdy mikrokontroler z tej rodziny posiada dwie grupy ustawianych bitów. Są to bity blokujące i bezpieczniki. Nazwa *bezpiecznik* jest w przypadku AVR bardzo myląca, ponieważ ustawienie bezpiecznika jest odwracalne. Daje się nim manipulować podobnie jak nastawami bitów blokujących.

W momencie zakupu układu od producenta zarówno bezpieczniki, jak i bity blokujące znajdują się w stanie logicznym wysokim (1). Programowanie ich polega na zmianie stanu z wysokiego na niski (0). Zarówno jedna, jak i druga grupa opcji nie jest umieszczona w normalnie dostępnym obszarze pamięci EEPROM czy też Flash. Mogą one być jednak ustawiane z poziomu aplikacji, za wyjątkiem bitów odpowia-

Tab. 3. Funkcje bezpieczników

Nazwa bezpiecznika	Opis funkcji
RCEN	RC Oscillator Enable Poprzez ustawienie tego bezpiecznika wewnętrzny oscylator RC może być włączony jako główny zegar mikrokontrolera. Niektóre AVR są dostarczane z tym bezpiecznikiem ustawionym, inne nie. Dlatego też należy zawsze sprawdzać domyślny stan tego bezpiecznika w odpowiedniej karcie katalogowej albo też ustawić go tuż po zakupie.
SPIEN	Serial Programming Interface Enable Ustawienie bezpiecznika umożliwia programowanie układu poprzez interfejs SPI. Jeśli bezpiecznik jest skasowany, interfejs SPI nie funkcjonuje i nie jest możliwe programowanie układu „in-system”.
RSTDISBL	Reset Disable Ustawienie powoduje zmianę funkcji wyprowadzenia RESET. W niektórych AVR wyprowadzenie RESET zmieniane jest w typową linię wejścia / wyjścia odpowiedniego portu, w innych w linię tylko wyjściową. Szczegóły zawsze podane są w odpowiedniej karcie katalogowej.
FSTRT	Fast Start Bezpiecznik kontroluje czas, po jakim startuje CPU mikrokontrolera od momentu startu oscylatora. Jeśli używany jest rezonator ceramiczny lub szybko uruchamiany zewnętrzny generator zegarowy, można ustawić ten bit. Umożliwi to CPU szybsze uruchomienie programu.
BODEN	Brown-Out Detection Enable Poprzez ustawienie zezwala się na pracę wewnętrznego układu kontroli napięcia zasilania. Jeśli napięcie nie mieści się w granicach określonych jako właściwe dla poprawnej pracy układu, generowany jest wewnętrzny sygnał <i>reset</i> .
BODLEVEL	Brown-Out Detection Level Bezpiecznik ten zmienia napięcie zadziałania układu kontroli napięcia zasilania oraz czas startu CPU. Oba te parametry są zależne od typu układu - szczegółów dotyczących działania należy szukać w danych katalogowych konkretnego mikrokontrolera.
CKSEL n..0	Clock Select Stan bezpieczników CKSEL kontroluje ustawienia rodzaju oscylatora, a w niektórych AVR również czas startu CPU. Rolę tych bezpieczników dla danego modelu mikrokontrolera opisuje jego karta katalogowa.
EESAVE	EEPROM Save Ustawienie bezpiecznika powoduje, że zawartość pamięci EEPROM nie jest kasowana w momencie programowania układu. W przypadku ustawienia tego bezpiecznika, aby skasować EEPROM, konieczne jest jego wyzerowanie.
SUT n..0	Start-Up Time Stan bezpiecznika umożliwia nastawę opóźnienia pomiędzy zanikiem zewnętrznego sygnału <i>reset</i> ałączeniem CPU mikrokontrolera. Ten czas może być wybrany w zależności np. od rodzaju użytego generatora zegarowego. Jeśli dla przykładu używany jest zewnętrzny generator zegarowy, który uruchamia się bardzo szybko - czas ten może być bardzo krótki. Gdy natomiast stosuje się rezonator kwarcowy - wymagany jest dłuższy czas na stabilizację generowanej częstotliwości.
BOOTRST	Boot Reset Ustawienie bezpiecznika decyduje, pod który adres wykona skok CPU mikrokontrolera po sygnale <i>reset</i> - czy wykona skok do fragmentu kodu umieszczonego w tablicy wektorów przerwań jako wektor 0, czy też do umieszczonego w obszarze tak zwanego Boot Loadera. Stan logiczny wysoki powoduje umieszczenie wektora przerwania po <i>reset</i> w obszarze sekcji Boot Loadera: będzie on wskazywał adres zależny od stanu BOOTSZ. Niektóre z mikrokontrolerów AVR mają możliwość umieszczenia wektora przerwania po <i>reset</i> zarówno w obszarze Boot Loadera, jak również pamięci aplikacji. Lokalizacja tego wektora zależna jest od stanu bitu IVSEL w rejestrze GICR.
BOOTSZ n..0	Boot Size Stan bezpieczników określa rozmiar i adres startowy bloku Boot Loadera. Sekcja Boot jest dostępna tylko i wyłącznie w mikrokontrolerach AVR, które mają w swojej liście rozkazów instrukcję SPM.
INTCAP	Internal Capacitors Gdy INCAP jest ustawiony, nie jest konieczne stosowanie zewnętrznych kondensatorów dołączanych do rezonatora kwarcowego. Obniża to koszt systemu i upraszcza płytke drukowaną.

dających za włączenie opcji tzw. *Boot Loader* w mikroprocesorach, gdzie jest możliwe „samoprogramowanie”. Jednak dla praktycznych zastosowań największe znaczenie będzie miało to, że nastawy bitów blokujących i bezpieczników mogą być zmieniane w czasie programowania, przy

pomocy programatora szeregowego albo równoległego. Ten drugi oferuje jednak dostęp do większej liczby nastaw.

Bity blokujące *lock bits*

Wszystkie mikrokontrolery AVR posiadają dwa bity blokujące o nazwie LB1 i LB2. Zmia-

na stanu tych bitów powoduje zabezpieczenie zawartości pamięci EEPROM i Flash. Zabezpieczenie podzielone jest na trzy poziomy (Mode 1...3), gdzie poziom 1 wyłącza zabezpieczenia, natomiast poziom 3 oferuje maksymalny jego stopień dla danego modelu. Możliwa jest zmiana poziomu zabezpieczenia poprzez zmianę stanów *lock bits*. I tu jedna bardzo ważna uwaga: w obrębie rodziny mikrokontrolerów AVR możliwa jest wyłącznie zmiana stanu bitu zabezpieczenia z wysokiego na niski, nigdy zaś odwrotnie. Przejście na niższy poziom zabezpieczenia zawsze wiąże się ze skasowaniem pamięci programu (Flash) układu. W ten prosty sposób możliwe jest zwiększanie poziomu zabezpieczeń, nigdy zaś zmniejszanie - wiąże się to bowiem z utratą danych.

Dodatkowo mikrokontrolery AVR, mające możliwość samoprogramowania, posiadają jeszcze

cztery dodatkowe bity zabezpieczeń BLB01, BLB02, BLB11 i BLB12. Ich ustawienie/kasowanie wiąże się z pewnymi restrykcjami dla instrukcji LPM (*Load Program Memory*) i SPM (*Store Program Memory*), jednak w przypadku specyficznych nastaw tych bitów proszę odnieść się do konkretnych kart katalogowych danego modelu mikrokontrolera.

Bezpieczniki *fuses*

Bity blokujące dostępne są we wszystkich trybach programowania. W przypadku bezpieczników jest inaczej. Większość z nich jest dostępna we wszystkich trybach programowania, natomiast niektóre z nich tylko w wybranych. Również inaczej niż w przypadku *lock bits*, stan bezpieczników nie ulega zmianie po instrukcji kasowania układu. Aby zmienić stan bezpiecznika, wymagane jest zaprogramowanie logicznego stanu wysokiego we właściwej lokalizacji pamięci. Taka zmiana stanu

bezpiecznika będzie odczuwalna dopiero po wykonaniu przez mikrokontroler wewnętrznej procedury obsługi sygnału *reset* generowanego po włączeniu zasilania (*power on reset*). Bardzo ważną informacją jest, że po ustawieniu bitów LB w trybie 2 lub 3 nie jest możliwa zmiana stanu żadnego z bezpieczników, zabroniony jest bowiem zapis do pamięci mikrokontrolera.

W **tab. 2** zawarto informacje o tym, jakie bezpieczniki i bity blokujące dostępne są w różnych mikrokontrolerach z rodziny AVR. Informuje również, w jakim trybie programowania możliwa jest zmiana ich nastaw. **Tab. 3** podaje natomiast, jaka jest rola ustawień poszczególnych bezpieczników.

Jacek Bogusz, AVT
jacek.bogusz@ep.com.pl

Dodatkowe informacje

- <http://www.avrfreaks.net/>
- <http://www.atmel.com/>