Жүйелерді FPGA негізінде жобалау

Дәріс 6: FPGA негізінде соңғы күй автоматтарын жобалау

ДӘРІСКЕР: PhD, доцент м.а.
Сқабылов Әлішер Әлиұлы

Дәрістің мақсаты:

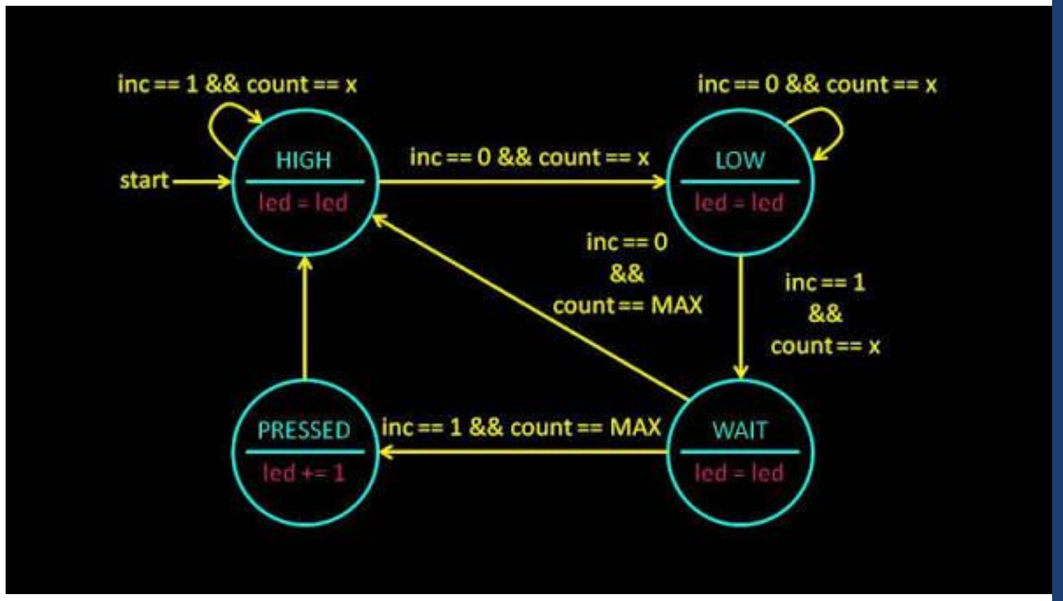
Соңғы күй автоматтарының теориялық негіздерін меңгеру, олардың цифрлық жүйелердегі қызметін түсіндіру, Moore және Mealy автоматтарының ерекшеліктерін қарастыру және FPGA негізіндегі жүйелерде соңғы күй автоматтарын HDL тілдері арқылы жобалау принциптерін үйрету.

Соңғы күй автоматтары (FSM)

FSM (Finite State Machine) — жүйенің жұмысын шектеулі күйлер жиыны және сол күйлер арасындағы ауысу ережелері арқылы сипаттайтын цифрлық басқару моделі.

Әрбір күй жүйенің белгілі бір жағдайын білдіреді. Кіріс сигналдары мен басқару шарттарына байланысты автомат келесі күйге өтеді және тиісті шығыс сигналдарын қалыптастырады.

FSM бағдаршамдарды басқаруда, байланыс протоколдарында, деректерді қабылдау және өңдеу құрылғыларында қолданылады. FPGA ішінде ол басқару алгоритмдерін аппараттық түрде іске асыруға мүмкіндік береді.



Мысал: IDLE → RECEIVE → CHECK → PROCESS → DONE.

FSM құрылымы: күй, кіріс, шығыс және ауысу

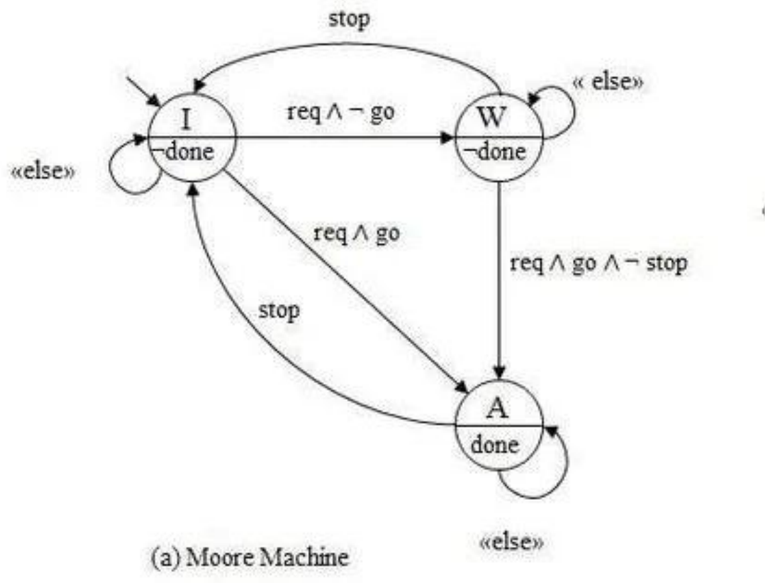
FSM **төрт негізгі элементтен** тұрады:

- *Күй* (State) — жүйенің ағымдағы жұмыс жағдайы. Мысалы, IDLE, RUN немесе DONE.
- *Кіріс* (Input) — автоматтың келесі күйін анықтауға әсер ететін сигнал. Мысалы, start, stop, ready немесе error.
- *Шығыс* (Output) — автоматтың күйіне немесе кіріс сигналдарына сәйкес қалыптасатын сигнал.
- *Ауысу* (State Transition) — бір күйден екінші күйге өту ережесі.

FSM жұмысының жалпы **құрылымы**:
Кіріс сигналдары → Келесі күй логикасы → Күй регистрі → Ағымдағы күй.
Шығыс логикасы ағымдағы күйді, ал Mealy моделінде кіріс сигналдарын да пайдаланады.

Moore және Mealy автоматтары

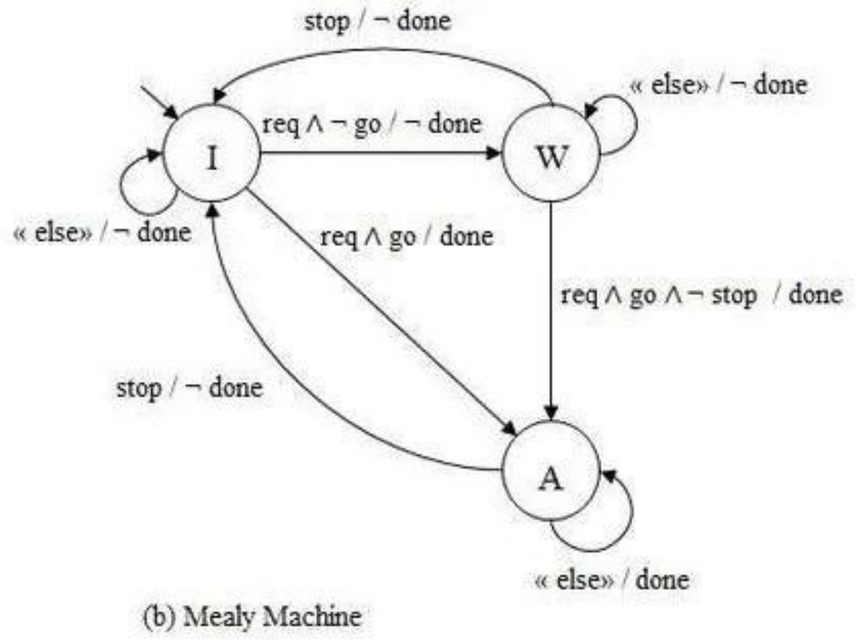
FSM-нің **екі негізгі түрі** — *Moore* және *Mealy* автоматтары.
Олардың басты айырмашылығы шығыс сигналдарының анықталу тәсілінде.
Moore автоматы: шығыс тек ағымдағы күйге тәуелді.
Output = f(Current State)
Бұл модельде шығыс күйге байланысты өзгереді. Шығыс логикасы салыстырмалы түрде қарапайым ұйымдастырылады.



(a) Moore Machine

Moore және Mealy автоматтары

Mealy автоматы: шығыс ағымдағы күйге және кіріс сигналдарына тәуелді.
Output = f(Current State, Input)
Бұл модель кіріс өзгерісіне жылдамырақ реакция беруі мүмкін және кейбір жағдайларда күйлер санын азайтады.
Модельді таңдау жобаның басқару талаптарына және шығыс сигналдарының қалай жұмыс істеуі керектігіне байланысты.



(b) Mealy Machine

Күй диаграммасы және күй кестесі

Күй диаграммасы (State Diagram) автоматтың күйлерін және олардың арасындағы ауысуларды графикалық түрде көрсетеді. Әр ауысу белгілі бір шартпен орындалуы мүмкін.
Мысалы: IDLE → RUN → DONE
RUN күйінен DONE күйіне өту үшін ready = 1 шарты қолданылуы мүмкін.

Күй кестесі (State Table) ағымдағы күйді, кіріс сигналын, келесі күйді және шығысты кесте түрінде көрсетеді.

Ағымдағы күй	Кіріс	Келесі кү
IDLE	start = 0	IDLE
IDLE	start = 1	RUN
RUN	done = 0	RUN

Күйлерді кодтау және күй регистрі

- FPGA ішінде автоматтың әр күйі белгілі бір кодпен ұсынылады. Бұл процесс **State Encoding** деп аталады.

- *Binary encoding*: күйлер екілік кодтармен белгіленеді. Мысалы, төрт күйге 00, 01, 10 және 11 кодтарын беруге болады. Бұл тәсілге екі бит жеткілікті.

- *One-hot encoding*: әр күйге жеке бит бөлінеді. Төрт күй үшін 0001, 0010, 0100 және 1000 кодтары қолданылуы мүмкін. Бұл тәсілде әр уақытта бір күйдің биті белсенді болады.

- *State Register* ағымдағы күйді сақтайды. Тактілік сигналдың тиісті фронтында регистр жаңа күйдің кодын қабылдайды. Кодтау тәсілі логикалық ресурстардың пайдаланылуына және автоматтың құрылымына әсер етуі мүмкін.

FSM-ді HDL тілінде сипаттау

SystemVerilog тілінде FSM сипаттамасын үш бөлікке ұйымдастыруға болады: күй анықтамасы, күй регистрі және келесі күйді анықтайтын логика.
Мысалы, автоматтың күйлері:

```
typedef enum logic [1:0] {
    IDLE, RUN, DONE
} state_t;

state_t current_state, next_state;
```

Күй регистрі:

```
always_ff @(posedge clk) begin
    if (reset)
        current_state <= IDLE;
    else
        current_state <= next_state;
end
```

Мұнда *always_ff* тактілік сигналға байланысты күйді сақтауға арналған. *always_comb* блогында ағымдағы күй мен кіріс сигналдары бойынша келесі күй есептеледі.

HDL сипаттамасы күй диаграммасы мен ауысу кестесіне сәйкес болуы керек.

Синхронды FSM және бастапқы күй

Синхронды FSM-де күй регистрі тактілік сигналға сәйкес жаңартылады. Келесі күйдің логикасы алдын ала есептелгенімен, жаңа күй регистрде тиісті тактілік оқиға кезінде ғана сақталады.

Reset сигналы автоматты анықталған бастапқы күйге орнатады. Мысалы, reset белсенді болған кезде автомат IDLE күйіне көшеді. Бастапқы күйді таңдау жүйенің қауіпсіз жұмысымен байланысты. Деректерді қабылдайтын контроллер WAIT күйінен, ал басқару құрылғысы IDLE күйінен басталуы мүмкін.

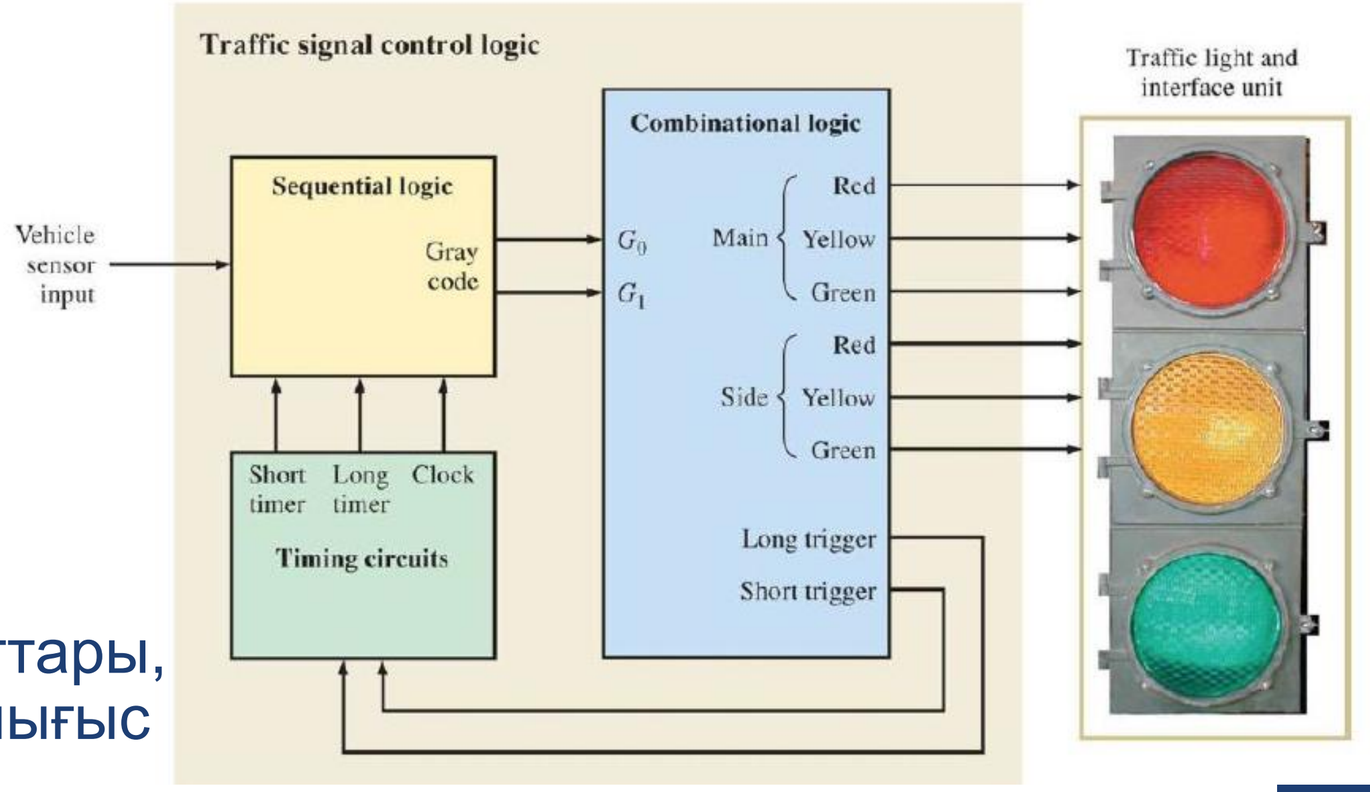
HDL сипаттамасында жарамсыз немесе күтпеген күй пайда болған жағдайда қауіпсіз күйге қайтару үшін default тармағын қарастырған жөн.

Практикалық мысал: бағдаршам контроллері

Бағдаршам контроллерін үш күйлі FSM арқылы жүзеге асыруға болады:
RED → GREEN → YELLOW → RED
Әр күйде тиісті жарық сигналы қосылады. Автомат келесі күйге timer_done сигналы белсенді болғанда өтеді. Бұл сигнал белгіленген уақыттың аяқталғанын білдіреді.

- RED күйінде қызыл жарық.
- GREEN күйінде жасыл жарық.
- YELLOW күйінде сары жарық.

Уақытты өлшейтін санауыш жеке аппараттық блок ретінде ұйымдастырылуы мүмкін. FSM осы санауыштың аяқталу сигналын пайдаланып, келесі күйді таңдайды. Жобалау кезінде күйлер, ауысу шарттары, бастапқы күй және әр күйге сәйкес шығыс сигналдары алдын ала анықталады.



FSM жобалау кезеңдері және жиі кездесетін қателер

FSM жобалау кезеңдері:

1. Жүйенің жұмыс талаптарын анықтау.
2. Күйлер мен ауысу шарттарын белгілеу.
3. Күй диаграммасы мен күй кестесін құру.
4. Күйлерді кодтау.
5. HDL сипаттамасын жазу.
6. Simulation арқылы тексеру.
7. FPGA құрылғысына іске асыру.

Жобалау кезінде кейбір күйлердің ауысу ережелерін жазбай кету, кіріс комбинацияларын толық қарастырмау, reset күйін дұрыс таңдамау немесе HDL кодын күй диаграммасына сәйкес келтірмеу сияқты қателер кездеседі. **Simulation** кезінде reset жұмысы, әртүрлі кіріс комбинациялары, күй ауысулары және шығыс сигналдары тексеріледі. Бұл автоматтың белгіленген алгоритмге сай жұмыс істейтінін бағалауға мүмкіндік береді.

Білімді бекітуге арналған сұрақтар:

1. Соңғы күй автоматы (FSM) дегеніміз не және ол қандай мақсатта қолданылады?
2. FSM құрылымының негізгі элементтерін атаңыз.
3. Күй (State) мен ауысу (State Transition) ұғымдарының айырмашылығы қандай?
4. Moore және Mealy автоматтарының негізгі айырмашылығы неде?
5. Күй диаграммасы мен күй кестесі қандай мақсатта қолданылады?
6. Binary және One-hot кодтау тәсілдерінің айырмашылығы қандай?
7. State Register қандай қызмет атқарады?
8. FSM-ді HDL тілінде сипаттауда always_ff және always_comb конструкциялары қалай қолданылады?
9. Reset сигналы FSM жұмысына не үшін қажет?
10. Синхронды FSM-де келесі күй қашан ағымдағы күйге айналады?
11. Бағдаршам контроллерін FSM арқылы қалай ұйымдастыруға болады?
12. FSM жобалау кезінде қандай қателер кездеседі және оларды Simulation арқылы қалай анықтауға болады?

Әдебиеттер тізімі:

1. Brown S., Vranesic Z. Fundamentals of Digital Logic with Verilog Design. McGraw-Hill.
2. Harris D., Harris S. Digital Design and Computer Architecture. Morgan Kaufmann.
3. Wakerly J. F. Digital Design: Principles and Practices. Pearson.
4. Mano M. M., Ciletti M. D. Digital Design. Pearson.
5. Roth C. H., Kinney L. L. Fundamentals of Logic Design. Cengage Learning.
6. Chu P. P. FPGA Prototyping by Verilog Examples. Wiley.
7. IEEE. IEEE Standard for SystemVerilog — Unified Hardware Design, Specification, and Verification Language. IEEE.
8. AMD. Vivado Design Suite User Guide: Synthesis. AMD Documentation.
9. AMD. Vivado Design Suite User Guide: Design Analysis and Closure Techniques. AMD Documentation.
10. Pedroni V. A. Circuit Design and Simulation with VHDL. MIT Press.