



Introduction to Digital Design

Week 11: Register-Transfer Level Design

Yao Zheng
University of Hawai'i at Mānoa
Department of Electrical and Computer Engineering

1

Overview

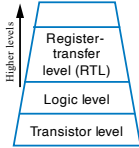
- RTL Introduction
 - Modern digital design involves creating processor-level components
 - High-level state machines
- RTL design process
 - Capture behavior: Use HLASM
 - Convert to circuit: A. Create datapath B. Connect DP to controller C. Derive controller FSM
- More RTL design
 - More components, arrays, timers, control vs. data dominated
- Determining fastest clock frequency
 - By finding critical path
- Behavioral-level design – C to gates
 - By using method to convert C (subset) to high-level state machine

2

Introduction

5.1

- Week 1 – Week 5
 - Capture Comb. behavior: Equations, truth tables
 - Convert to circuit: AND + OR + NOT → Comb. logic
- Week 7 – Week 8
 - Capture sequential behavior: FSMs
 - Convert to circuit: Register + Comb. logic → Controller
- Week 9
 - Datapath components, simple datapaths
- Week 11 – Week 12
 - Capture behavior: High-level state machine
 - Convert to circuit: Controller + Datapath → Processor
 - Known as "RTL" (register-transfer level) design



Levels of digital design abstraction

Processors:

- Programmable (microprocessor)
- Custom

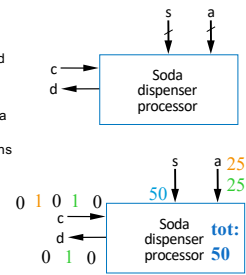
Note: Slides with animation are denoted with a small red "a" near the animated items

3

High-Level State Machines (HLSMs)

5.2

- Some behaviors too complex for equations, truth tables, or FSMs
- Ex: Soda dispenser
 - c : bit input, 1 when coin deposited
 - a : 8-bit input having value of deposited coin
 - s : 8-bit input having cost of a soda
 - d : bit output, processor sets to 1 when total value of deposited coins equals or exceeds cost of a soda
- FSM can't represent...
 - 8-bit input/output
 - Storage of current total
 - Addition (e.g., 25 + 10)



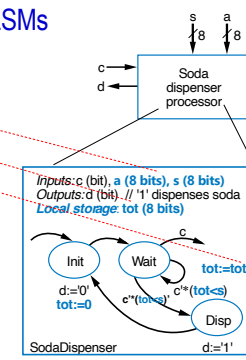
4



5

HLSMs

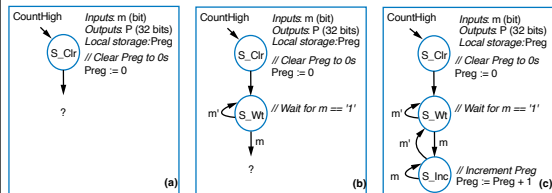
- High-level state machine (HLSM) extends FSM with:
 - Multi-bit input/output
 - Local storage
 - Arithmetic operations
- Conventions
 - Numbers:
 - Single-bit: '0' (single quotes)
 - Integer: 0 (no quotes)
 - Multi-bit: "0000" (double quotes)
 - == for equal, := for assignment
 - Multi-bit outputs *must* be registered via local storage
 - // precedes a comment



6

Ex: Cycles-High Counter

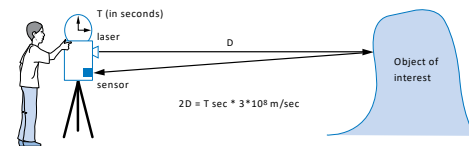
- P = total number (in binary) of cycles that m is 1
- Capture behavior as HLSM
 - Preg required (multibit outputs must be registered)
 - Use to hold count



Note: Could have designed directly using an up-counter. But, that methodology is ad hoc, and won't work for more complex examples, like the next one.

7

Example: Laser-Based Distance Measurer



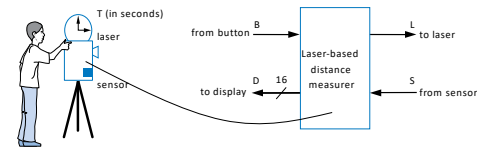
- Laser-based distance measurement – pulse laser, measure time T to sense reflection
 - Laser light travels at speed of light, $3 \cdot 10^8$ m/sec
 - Distance is thus $D = (T \text{ sec} \cdot 3 \cdot 10^8 \text{ m/sec}) / 2$

8



9

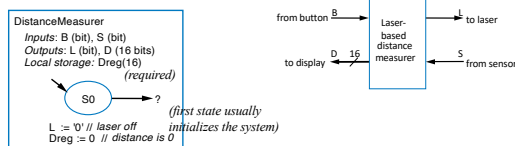
Example: Laser-Based Distance Measurer



- Inputs/outputs
 - B : bit input, from button, to begin measurement
 - L : bit output, activates laser
 - S : bit input, senses laser reflection
 - D : 16-bit output, to display computed distance

10

Example: Laser-Based Distance Measurer

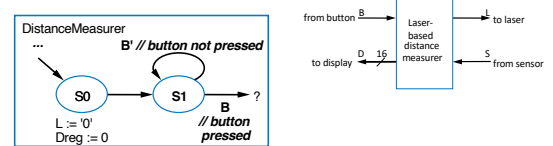


- Declare inputs, outputs, and local storage
 - Dreg required for multi-bit output
- Create initial state, name it **S0**
 - Initialize laser to off ($L := '0'$)
 - Initialize displayed distance to 0 ($Dreg := 0$)

Recall: '0' means single bit, 0 means integer

11

Example: Laser-Based Distance Measurer

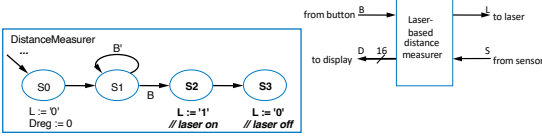


- Add another state, **S1**, that waits for a button press
 - B' – stay in **S1**, keep waiting
 - B – go to a new state **S2**

Q: What should **S2** do? A: Turn on the laser

12

Example: Laser-Based Distance Measurer

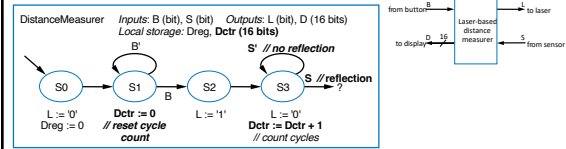


- Add a state **S2** that turns on the laser ($L:=1$)
- Then turn off laser ($L:=0$) in a state **S3**

Q: What do next? A: Start timer, wait to sense reflection

13

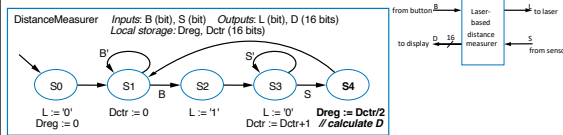
Example: Laser-Based Distance Measurer



- Stay in **S3** until sense reflection (S)
- To measure time, count cycles while in **S3**
 - To count, declare local storage *Dctr*
 - Initialize *Dctr* to 0 in **S1**. In **S2** would have been O.K. too.
 - Don't forget to initialize local storage—common mistake
 - Increment *Dctr* each cycle in **S3**

14

Example: Laser-Based Distance Measurer



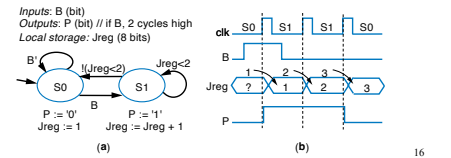
- Once reflection detected (S), go to new state **S4**
 - Calculate distance
 - Assuming clock frequency is 3×10^8 , *Dctr* holds number of meters, so $Dreg := Dctr/2$
- After **S4**, go back to **S1** to wait for button again

15

HLSM Actions: Updates Occur Next Clock Cycle

- Local storage updated on clock edges only
 - Enter state on clock edge
 - Storage writes in that state occur on *next* clock edge
 - Can think of as occurring on outgoing transitions
- **Thus**, transition conditions use the OLD value, not the newly-written value

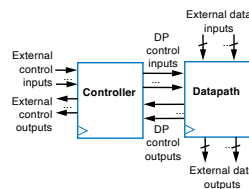
– Example:



16

RTL Design Process

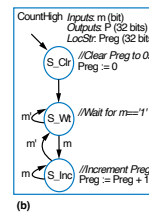
- Capture behavior
- Convert to circuit
 - Need target architecture
 - Datapath capable of HLSM's data operations
 - Controller to control datapath



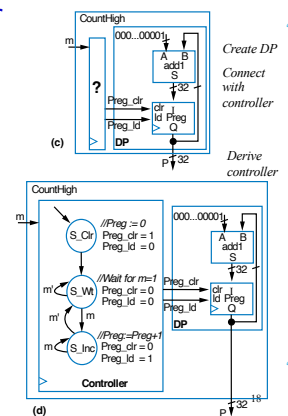
17

Ctrl/DP Example for Earlier Cycles-High Counter

First clear Preg to 0s
Then increment Preg for each clock cycle that m is 1



18



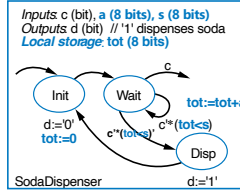
RTL Design Process

Step	Description
Step 1: Capture behavior	Capture a high-level state machine Describe the system's desired behavior as a high-level state machine. The state machine consists of states and transitions. The state machine is "high-level" because the transition conditions and the state actions are more than just Boolean operations on single-bit inputs and outputs.
2A	Create a datapath Create a datapath to carry out the data operations of the high-level state machine.
Step 2: Convert to circuit	2B Connect the datapath to a controller Connect the datapath to a controller block. Connect external control inputs and outputs to the controller block. 2C Derive the controller's FSM Convert the high-level state machine to a finite-state machine (FSM) for the controller, by replacing data operations with setting and reading of control signals to and from the datapath.

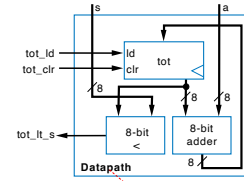
19

Example: Soda Dispenser from Earlier

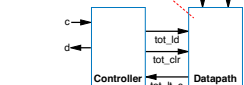
- Quick overview example. More details of each step to come.



Step 1



Step 2A

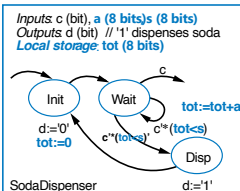


Step 2B

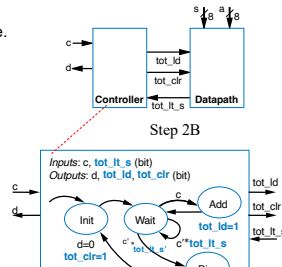
20

Example: Soda Dispenser

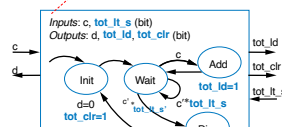
- Quick overview example. More details of each step to come.



Step 1



Step 2B

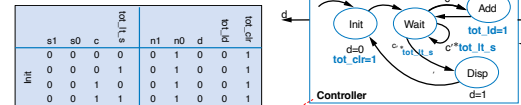


Step 2C

21

Example: Soda Dispenser

- Quick overview example. More details of each step to come.



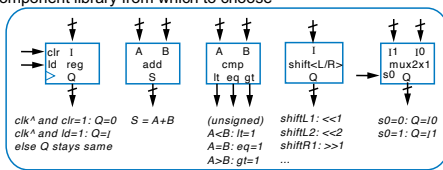
Step 2C

Use controller design process (Ch3) to complete the design

22

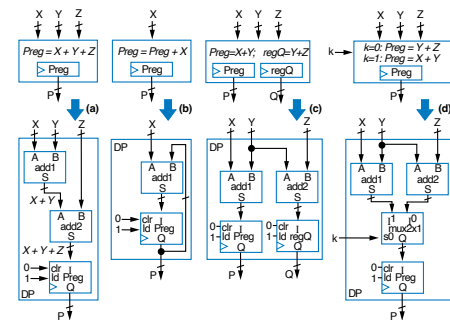
RTL Design Process—Step 2A: Create a datapath

- Sub-steps
 - HLSM data inputs/outputs → Datapath inputs/outputs.
 - HLSM local storage item → Instantiated register
 - "Instantiate": Add new component ("instance") to design
 - Each HLSM state action and transition condition data computation → Datapath components and connections
 - Also instantiate multiplexors as needed
- Need component library from which to choose



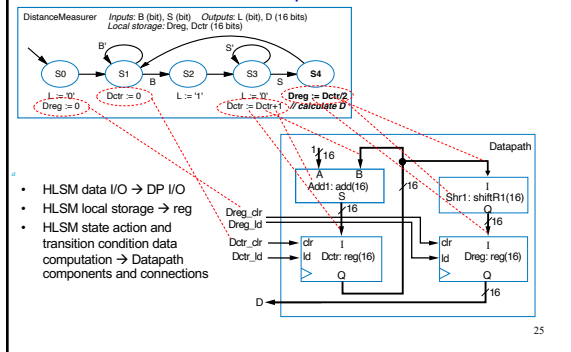
23

Step 2A: Create a Datapath—Simple Examples



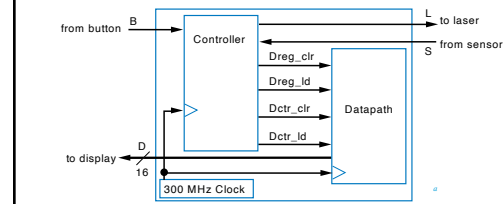
24

Laser-Based Distance Measurer—Step 2A: Create a Datapath



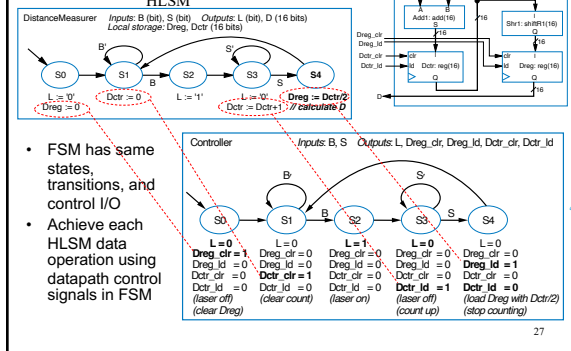
25

Laser-Based Distance Measurer—Step 2B: Connecting the Datapath to a Controller



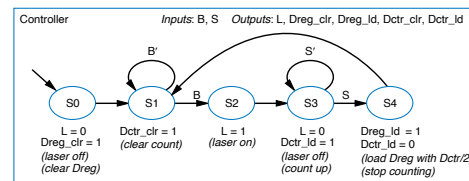
26

Laser-Based Distance Measurer—Step 2C: Derive the Controller FSM



27

Laser-Based Distance Measurer—Step 2C: Derive the Controller FSM



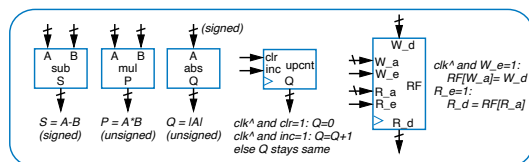
- Same FSM, using convention of unassigned outputs implicitly assigned 0

Some assignments to 0 still shown, due to their importance in understanding desired controller behavior

28

More RTL Design

- Additional datapath components



29

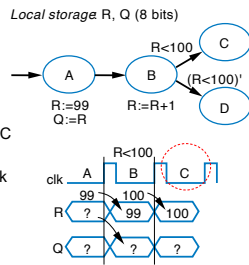
RTL Design Involving Register File or Memory

- HLSM array: Ordered list of items
 - Ex: Local storage: $A[4](8\text{-bit})$ – 4 8-bit items
 - Accessed using notation " $A[i]$ ", i is index
 - $A[0] := 9$; $A[1] := 8$; $A[2] := 7$; $A[3] := 22$
 - Array contents now: $\langle 9, 8, 7, 22 \rangle$
 - $X := A[1]$ will set X to 8
 - Note: First element's index is 0
- Array can be mapped to instantiated register file or memory

30

Common RTL Design Pitfall Involving Storage Updates

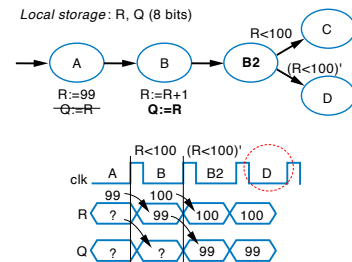
- Questions
 - Value of Q after state A?
 - Final state is C or D?
- Answers
 - Q is NOT 99 after state A
 - R is 99 in state B, so final state is C
 - Storage update actions in state occur *simultaneously* on next clock edge
 - Thus, order actions are written is irrelevant
 - A's actions same if:
 - $Q:=R$ $R:=99$ or
 - $R:=99$ $Q:=R$



37

Common RTL Design Pitfall Involving Storage Updates

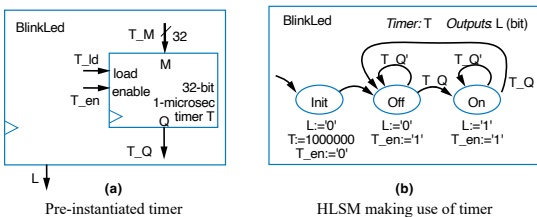
- New HLSM using extra state so read of R occurs after write of R



38

RTL Design Involving a Timer

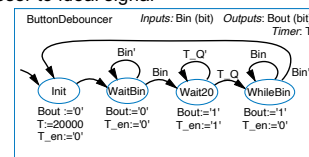
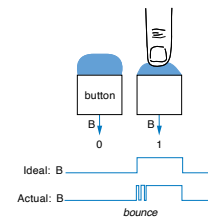
- Commonly need explicit time intervals
 - Ex: Repeatedly blink LED on 1 second, off 1 second
- Pre-instantiate timer that HLSM can then use



39

Button Debouncing

- Press button
 - Ideally, output changes to 1
 - Actually, output bounces
 - Due to mechanical reasons
 - Like ball bouncing when dropped to floor
- Digital circuit can convert actual signal closer to ideal signal



40

Data Dominated RTL Design Example

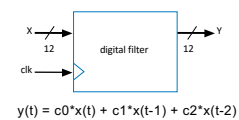
- Data dominated design: Extensive DP, simple controller
- Control dominated design: Complex controller, simple DP
- Example: Filter
 - Converts digital input stream to new digital output stream
 - Ex: Remove noise
 - 180, 180, 181, 180, 240, 180, 181
 - 240 is probably noise, filter might replace by 181
 - Simple filter: Output average of last N values
 - Small N : less filtering
 - Large N : more filtering, but less sharp output



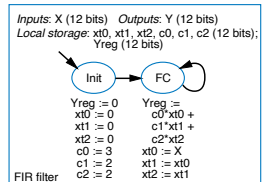
41

Data Dominated RTL Design Example: FIR Filter

- FIR filter
 - "Finite Impulse Response"
 - Simply a configurable weighted sum of past input values
 - $y(t) = c_0 \cdot x(t) + c_1 \cdot x(t-1) + c_2 \cdot x(t-2)$
 - Above known as "3 tap"
 - Tens of taps more common
 - Very general filter – User sets the constants (c_0 , c_1 , c_2) to define specific filter

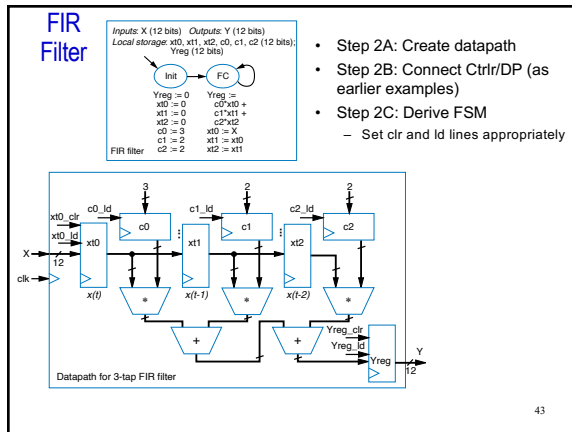


$$y(t) = c_0 \cdot x(t) + c_1 \cdot x(t-1) + c_2 \cdot x(t-2)$$

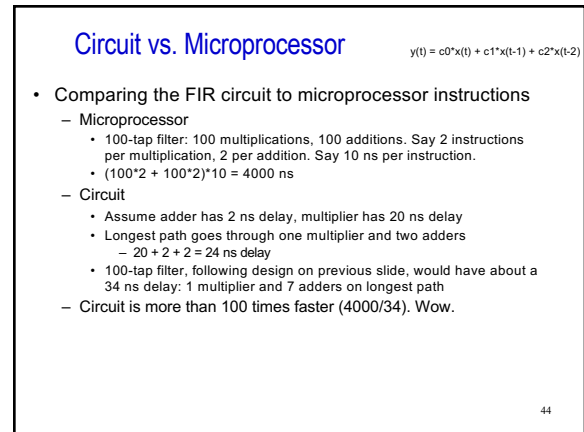


Assume constants set to 3, 2, and 2

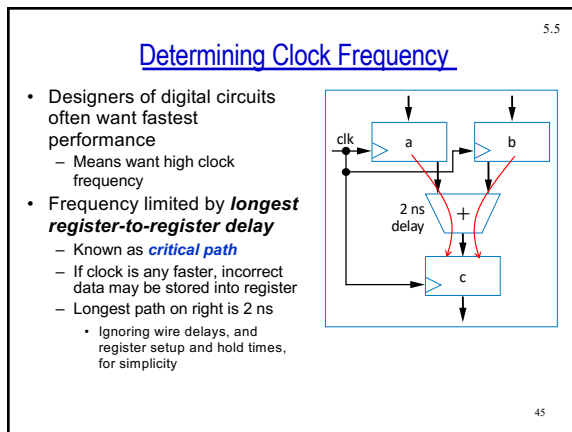
42



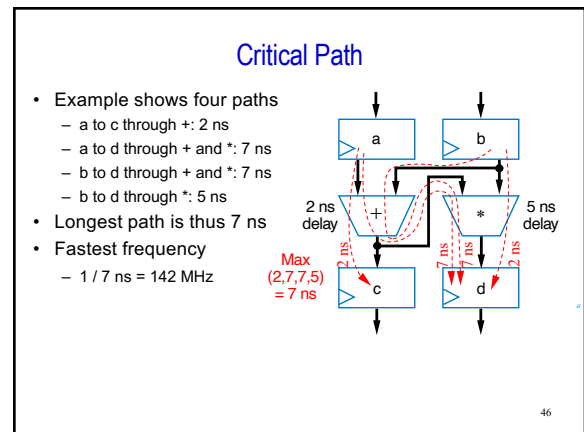
43



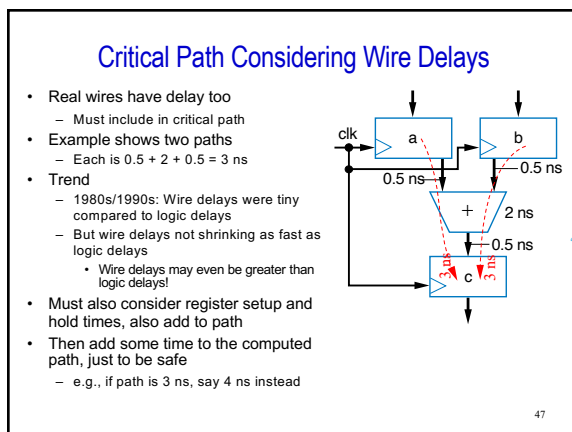
44



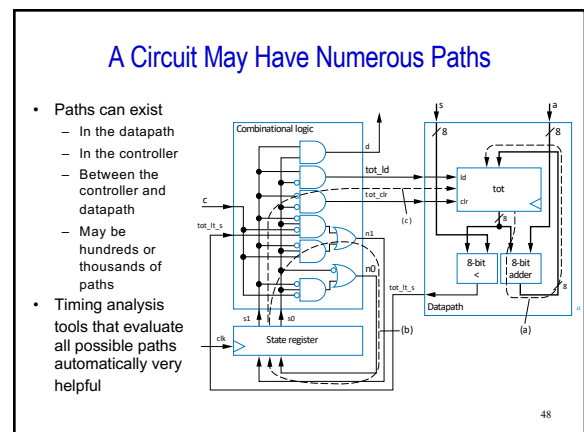
45



46



47



48

Behavioral Level Design: C to Gates

Inputs: A, B [256][8 bits]; go (bit)
Outputs: sad (32 bits)
Local storage: sum, sadreg (32 bits); i (8 bits)

C code

```
int SAD (byte A[256], byte B[256]) // not quite C syntax
{
    uint sum; short uint i;
    sum = 0;
    i = 0;
    while (i < 256) {
        sum = sum + abs(A[i] - B[i]);
        i = i + 1;
    }
    return sum;
}
```

Diagram showing the conversion of the C code to a high-level state machine (HLSM) with states S0, S1, S2, S3, and S4. The diagram is crossed out with a red X.

- Earlier sum-of-absolute-differences example
 - Started with high-level state machine
 - C code is an even better starting point -- easier to understand

49

Converting from C to High-Level State Machine

- Convert each C construct to equivalent states and transitions
- Assignment statement**
 - Becomes one state with assignment
- If-then statement**
 - Becomes state with condition check, transitioning to "then" statements if condition true, otherwise to ending state
 - "then" statements would also be converted to states

Diagram illustrating the conversion of an assignment statement and an if-then statement to states and transitions.

50

Converting from C to High-Level State Machine

- If-then-else**
 - Becomes state with condition check, transitioning to "then" statements if condition true, or to "else" statements if condition false
- While loop statement**
 - Becomes state with condition check, transitioning to while loop's statements if true, then transitioning back to condition check

Diagram illustrating the conversion of an if-then-else statement and a while loop statement to states and transitions.

51

Simple Example of Converting from C to High-Level State Machine

Inputs: uint X, Y
Outputs: uint Max

Diagram illustrating the conversion of a simple example (Computing the maximum of two numbers) to a high-level state machine.

- Simple example: Computing the maximum of two numbers
 - Convert if-then-else statement to states (b)
 - Then convert assignment statements to states (c)

52

Example: SAD C code to HLSM

Inputs: byte A[256], B[256]
Output: int sad
Local storage: sum, sadreg (32 bits); i (8 bits)

Diagram illustrating the conversion of the SAD C code to a high-level state machine (HLSM) with states S0, S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15, S16, S17, S18, S19, S20, S21, S22, S23, S24, S25, S26, S27, S28, S29, S30, S31, S32, S33, S34, S35, S36, S37, S38, S39, S40, S41, S42, S43, S44, S45, S46, S47, S48, S49, S50, S51, S52, S53, S54, S55, S56, S57, S58, S59, S60, S61, S62, S63, S64, S65, S66, S67, S68, S69, S70, S71, S72, S73, S74, S75, S76, S77, S78, S79, S80, S81, S82, S83, S84, S85, S86, S87, S88, S89, S90, S91, S92, S93, S94, S95, S96, S97, S98, S99, S100, S101, S102, S103, S104, S105, S106, S107, S108, S109, S110, S111, S112, S113, S114, S115, S116, S117, S118, S119, S120, S121, S122, S123, S124, S125, S126, S127, S128, S129, S130, S131, S132, S133, S134, S135, S136, S137, S138, S139, S140, S141, S142, S143, S144, S145, S146, S147, S148, S149, S150, S151, S152, S153, S154, S155, S156, S157, S158, S159, S160, S161, S162, S163, S164, S165, S166, S167, S168, S169, S170, S171, S172, S173, S174, S175, S176, S177, S178, S179, S180, S181, S182, S183, S184, S185, S186, S187, S188, S189, S190, S191, S192, S193, S194, S195, S196, S197, S198, S199, S200, S201, S202, S203, S204, S205, S206, S207, S208, S209, S210, S211, S212, S213, S214, S215, S216, S217, S218, S219, S220, S221, S222, S223, S224, S225, S226, S227, S228, S229, S230, S231, S232, S233, S234, S235, S236, S237, S238, S239, S240, S241, S242, S243, S244, S245, S246, S247, S248, S249, S250, S251, S252, S253, S254, S255, S256, S257, S258, S259, S260, S261, S262, S263, S264, S265, S266, S267, S268, S269, S270, S271, S272, S273, S274, S275, S276, S277, S278, S279, S280, S281, S282, S283, S284, S285, S286, S287, S288, S289, S290, S291, S292, S293, S294, S295, S296, S297, S298, S299, S300, S301, S302, S303, S304, S305, S306, S307, S308, S309, S310, S311, S312, S313, S314, S315, S316, S317, S318, S319, S320, S321, S322, S323, S324, S325, S326, S327, S328, S329, S330, S331, S332, S333, S334, S335, S336, S337, S338, S339, S340, S341, S342, S343, S344, S345, S346, S347, S348, S349, S350, S351, S352, S353, S354, S355, S356, S357, S358, S359, S360, S361, S362, S363, S364, S365, S366, S367, S368, S369, S370, S371, S372, S373, S374, S375, S376, S377, S378, S379, S380, S381, S382, S383, S384, S385, S386, S387, S388, S389, S390, S391, S392, S393, S394, S395, S396, S397, S398, S399, S400, S401, S402, S403, S404, S405, S406, S407, S408, S409, S410, S411, S412, S413, S414, S415, S416, S417, S418, S419, S420, S421, S422, S423, S424, S425, S426, S427, S428, S429, S430, S431, S432, S433, S434, S435, S436, S437, S438, S439, S440, S441, S442, S443, S444, S445, S446, S447, S448, S449, S450, S451, S452, S453, S454, S455, S456, S457, S458, S459, S460, S461, S462, S463, S464, S465, S466, S467, S468, S469, S470, S471, S472, S473, S474, S475, S476, S477, S478, S479, S480, S481, S482, S483, S484, S485, S486, S487, S488, S489, S490, S491, S492, S493, S494, S495, S496, S497, S498, S499, S500, S501, S502, S503, S504, S505, S506, S507, S508, S509, S510, S511, S512, S513, S514, S515, S516, S517, S518, S519, S520, S521, S522, S523, S524, S525, S526, S527, S528, S529, S530, S531, S532, S533, S534, S535, S536, S537, S538, S539, S540, S541, S542, S543, S544, S545, S546, S547, S548, S549, S550, S551, S552, S553, S554, S555, S556, S557, S558, S559, S560, S561, S562, S563, S564, S565, S566, S567, S568, S569, S570, S571, S572, S573, S574, S575, S576, S577, S578, S579, S580, S581, S582, S583, S584, S585, S586, S587, S588, S589, S590, S591, S592, S593, S594, S595, S596, S597, S598, S599, S600, S601, S602, S603, S604, S605, S606, S607, S608, S609, S610, S611, S612, S613, S614, S615, S616, S617, S618, S619, S620, S621, S622, S623, S624, S625, S626, S627, S628, S629, S630, S631, S632, S633, S634, S635, S636, S637, S638, S639, S640, S641, S642, S643, S644, S645, S646, S647, S648, S649, S650, S651, S652, S653, S654, S655, S656, S657, S658, S659, S660, S661, S662, S663, S664, S665, S666, S667, S668, S669, S670, S671, S672, S673, S674, S675, S676, S677, S678, S679, S680, S681, S682, S683, S684, S685, S686, S687, S688, S689, S690, S691, S692, S693, S694, S695, S696, S697, S698, S699, S700, S701, S702, S703, S704, S705, S706, S707, S708, S709, S710, S711, S712, S713, S714, S715, S716, S717, S718, S719, S720, S721, S722, S723, S724, S725, S726, S727, S728, S729, S730, S731, S732, S733, S734, S735, S736, S737, S738, S739, S740, S741, S742, S743, S744, S745, S746, S747, S748, S749, S750, S751, S752, S753, S754, S755, S756, S757, S758, S759, S760, S761, S762, S763, S764, S765, S766, S767, S768, S769, S770, S771, S772, S773, S774, S775, S776, S777, S778, S779, S780, S781, S782, S783, S784, S785, S786, S787, S788, S789, S790, S791, S792, S793, S794, S795, S796, S797, S798, S799, S800, S801, S802, S803, S804, S805, S806, S807, S808, S809, S810, S811, S812, S813, S814, S815, S816, S817, S818, S819, S820, S821, S822, S823, S824, S825, S826, S827, S828, S829, S830, S831, S832, S833, S834, S835, S836, S837, S838, S839, S840, S841, S842, S843, S844, S845, S846, S847, S848, S849, S850, S851, S852, S853, S854, S855, S856, S857, S858, S859, S860, S861, S862, S863, S864, S865, S866, S867, S868, S869, S870, S871, S872, S873, S874, S875, S876, S877, S878, S879, S880, S881, S882, S883, S884, S885, S886, S887, S888, S889, S890, S891, S892, S893, S894, S895, S896, S897, S898, S899, S900, S901, S902, S903, S904, S905, S906, S907, S908, S909, S910, S911, S912, S913, S914, S915, S916, S917, S918, S919, S920, S921, S922, S923, S924, S925, S926, S927, S928, S929, S930, S931, S932, S933, S934, S935, S936, S937, S938, S939, S940, S941, S942, S943, S944, S945, S946, S947, S948, S949, S950, S951, S952, S953, S954, S955, S956, S957, S958, S959, S960, S961, S962, S963, S964, S965, S966, S967, S968, S969, S970, S971, S972, S973, S974, S975, S976, S977, S978, S979, S980, S981, S982, S983, S984, S985, S986, S987, S988, S989, S990, S991, S992, S993, S994, S995, S996, S997, S998, S999, S1000, S1001, S1002, S1003, S1004, S1005, S1006, S1007, S1008, S1009, S1010, S1011, S1012, S1013, S1014, S1015, S1016, S1017, S1018, S1019, S1020, S1021, S1022, S1023, S1024, S1025, S1026, S1027, S1028, S1029, S1030, S1031, S1032, S1033, S1034, S1035, S1036, S1037, S1038, S1039, S1040, S1041, S1042, S1043, S1044, S1045, S1046, S1047, S1048, S1049, S1050, S1051, S1052, S1053, S1054, S1055, S1056, S1057, S1058, S1059, S1060, S1061, S1062, S1063, S1064, S1065, S1066, S1067, S1068, S1069, S1070, S1071, S1072, S1073, S1074, S1075, S1076, S1077, S1078, S1079, S1080, S1081, S1082, S1083, S1084, S1085, S1086, S1087, S1088, S1089, S1090, S1091, S1092, S1093, S1094, S1095, S1096, S1097, S1098, S1099, S1100, S1101, S1102, S1103, S1104, S1105, S1106, S1107, S1108, S1109, S1110, S1111, S1112, S1113, S1114, S1115, S1116, S1117, S1118, S1119, S1120, S1121, S1122, S1123, S1124, S1125, S1126, S1127, S1128, S1129, S1130, S1131, S1132, S1133, S1134, S1135, S1136, S1137, S1138, S1139, S1140, S1141, S1142, S1143, S1144, S1145, S1146, S1147, S1148, S1149, S1150, S1151, S1152, S1153, S1154, S1155, S1156, S1157, S1158, S1159, S1160, S1161, S1162, S1163, S1164, S1165, S1166, S1167, S1168, S1169, S1170, S1171, S1172, S1173, S1174, S1175, S1176, S1177, S1178, S1179, S1180, S1181, S1182, S1183, S1184, S1185, S1186, S1187, S1188, S1189, S1190, S1191, S1192, S1193, S1194, S1195, S1196, S1197, S1198, S1199, S1200, S1201, S1202, S1203, S1204, S1205, S1206, S1207, S1208, S1209, S1210, S1211, S1212, S1213, S1214, S1215, S1216, S1217, S1218, S1219, S1220, S1221, S1222, S1223, S1224, S1225, S1226, S1227, S1228, S1229, S1230, S1231, S1232, S1233, S1234, S1235, S1236, S1237, S1238, S1239, S1240, S1241, S1242, S1243, S1244, S1245, S1246, S1247, S1248, S1249, S1250, S1251, S1252, S1253, S1254, S1255, S1256, S1257, S1258, S1259, S1260, S1261, S1262, S1263, S1264, S1265, S1266, S1267, S1268, S1269, S1270, S1271, S1272, S1273, S1274, S1275, S1276, S1277, S1278, S1279, S1280, S1281, S1282, S1283, S1284, S1285, S1286, S1287, S1288, S1289, S1290, S1291, S1292, S1293, S1294, S1295, S1296, S1297, S1298, S1299, S1300, S1301, S1302, S1303, S1304, S1305, S1306, S1307, S1308, S1309, S1310, S1311, S1312, S1313, S1314, S1315, S1316, S1317, S1318, S1319, S1320, S1321, S1322, S1323, S1324, S1325, S1326, S1327, S1328, S1329, S1330, S1331, S1332, S1333, S1334, S1335, S1336, S1337, S1338, S1339, S1340, S1341, S1342, S1343, S1344, S1345, S1346, S1347, S1348, S1349, S1350, S1351, S1352, S1353, S1354, S1355, S1356, S1357, S1358, S1359, S1360, S1361, S1362, S1363, S1364, S1365, S1366, S1367, S1368, S1369, S1370, S1371, S1372, S1373, S1374, S1375, S1376, S1377, S1378, S1379, S1380, S1381, S1382, S1383, S1384, S1385, S1386, S1387, S1388, S1389, S1390, S1391, S1392, S1393, S1394, S1395, S1396, S1397, S1398, S1399, S1400, S1401, S1402, S1403, S1404, S1405, S1406, S1407, S1408, S1409, S1410, S1411, S1412, S1413, S1414, S1415, S1416, S1417, S1418, S1419, S1420, S1421, S1422, S1423, S1424, S1425, S1426, S1427, S1428, S1429, S1430, S1431, S1432, S1433, S1434, S1435, S1436, S1437, S1438, S1439, S1440, S1441, S1442, S1443, S1444, S1445, S1446, S1447, S1448, S1449, S1450, S1451, S1452, S1453, S1454, S1455, S1456, S1457, S1458, S1459, S1460, S1461, S1462, S1463, S1464, S1465, S1466, S1467, S1468, S1469, S1470, S1471, S1472, S1473, S1474, S1475, S1476, S1477, S1478, S1479, S1480, S1481, S1482, S1483, S1484, S1485, S1486, S1487, S1488, S1489, S1490, S1491, S1492, S1493, S1494, S1495, S1496, S1497, S1498, S1499, S1500, S1501, S1502, S1503, S1504, S1505, S1506, S1507, S1508, S1509, S1510, S1511, S1512, S1513, S1514, S1515, S1516, S1517, S1518, S1519, S1520, S1521, S1522, S1523, S1524, S1525, S1526, S1527, S1528, S1529, S1530, S1531, S1532, S1533, S1534, S1535, S1536, S1537, S1538, S1539, S1540, S1541, S1542, S1543, S1544, S1545, S1546, S1547, S1548, S1549, S1550, S1551, S1552, S1553, S1554, S1555, S1556, S1557, S1558, S1559, S1560, S1561, S1562, S1563, S1564, S1565, S1566, S1567, S1568, S1569, S1570, S1571, S1572, S1573, S1574, S1575, S1576, S1577, S1578, S1579, S1580, S1581, S1582, S1583, S1584, S1585, S1586, S1587, S1588, S1589, S1590, S1591, S1592, S1593, S1594, S1595, S1596, S1597, S1598, S1599, S1600, S1601, S1602, S1603, S1604, S1605, S1606, S1607, S1608, S1609, S1610, S1611, S1612, S1613, S1614, S1615, S1616, S1617, S1618, S1619, S1620, S1621, S1622, S1623, S1624, S1625, S1626, S1627, S1628, S1629, S1630, S1631, S1632, S1633, S1634, S1635, S1636, S1637, S1638, S1639, S1640, S1641, S1642, S1643, S1644, S1645, S1646, S1647, S1648, S1649, S1650, S1651, S1652, S1653, S1654, S1655, S1656, S1657, S1658, S1659, S1660, S1661, S1662, S1663, S1664, S1665, S1666, S1667, S1668, S1669, S1670, S1671, S1672, S1673, S1674, S1675, S1676, S1677, S1678, S1679, S1680, S1681, S1682, S1683, S1684, S1685, S1686, S1687, S1688, S1689, S1690, S1691, S1692, S1693, S1694, S1695, S1696, S1697, S1698, S1699, S1700, S1701, S1702, S1703, S1704, S1705, S1706, S1707, S1708, S1709, S1710, S1711, S1712, S1713, S1714, S1715, S1716, S1717, S1718, S1719, S1720, S1721, S1722, S1723, S1724, S1725, S1726, S1727, S1728, S1729, S1730, S1731, S1732, S1733, S1734, S1735, S1736, S1737, S1738, S1739, S1740, S1741, S1742, S1743, S1744, S1745, S1746, S1747, S1748, S1749, S1750, S1751, S1752, S1753, S1754, S1755, S1756, S1757, S1758, S1759, S1760, S1761, S1762, S1763, S1764, S1765, S1766, S1767, S1768, S1769, S1770, S1771, S1772, S1773, S1774, S1775, S1776, S1777, S1778, S1779, S1780, S1781, S1782, S1783, S1784, S1785, S1786, S1787, S1788, S1789, S1790, S1791, S1792, S1793, S1794, S1795, S1796, S1797, S1798, S1799, S1800, S1801, S1802, S1803, S1804, S1805, S1806, S1807, S1808, S1809, S1810, S1811, S1812, S1813, S1814, S1815, S1816, S1817, S1818, S1819, S1820, S1821, S1822, S1823, S1824, S1825, S1826, S1827, S1828, S1829, S1830, S1831, S1832, S1833, S1834, S1835, S1836, S1837, S1838, S1839, S1840, S1841, S1842, S1843, S1844, S1845, S1846, S1847, S1848, S1849, S1850, S1851, S1852, S1853, S1854, S1855, S1856, S1857, S1858, S1859, S1860, S1861, S1862, S1863, S1864, S1865, S1866, S1867, S1868, S1869, S1870, S1871, S1872, S1873, S1874, S1875, S1876, S1877, S1878, S1879, S1880, S1881, S1882, S1883, S1884, S1885, S1886, S1887, S1888, S1889, S1890, S1891, S1892, S1893, S1894, S1895, S1896, S1897, S1898, S1899, S1900, S1901, S1902, S1903, S1904, S1905, S1906, S1907, S1908, S1909, S1910, S1911, S1912, S1913, S1914, S1915, S1916, S1917, S1918, S1919, S1920, S1921, S1922, S1923, S1924, S1925, S1926, S1927, S1928, S1929, S1930, S1931, S1932, S1933, S1934, S1935, S1936, S1937, S1938, S1939, S1940, S1941, S1942, S1943, S1944, S1945, S1946, S1947, S1948, S1949, S1950, S1951, S1952, S1953, S1954, S1955, S1956, S1957, S1958, S1959, S1960, S1961, S1962, S1963, S1964, S1965, S1966, S1967, S1968, S1969, S1970, S1971, S1972, S1973, S1974, S1975, S1976, S1977, S1978, S1979, S1980, S1981, S1982, S1983, S1984, S1985, S1986, S1987, S1988, S1989, S1990, S1991, S1992, S1993, S1994, S1995, S1996, S1997, S1998, S1999, S2000, S2001, S2002, S2003, S2004, S2005, S2006, S2007, S2008, S2009, S2010, S2011, S2012, S2013, S2014, S2015, S2016, S2017, S2018, S2019, S2020, S2021, S2022, S2023, S2024, S2025, S2026, S2027, S2028, S2029, S2030, S2031, S2032, S2033, S2034, S2035, S2036, S2037, S2038, S2039, S2040, S2041, S2042, S2043, S2044, S2045, S2046, S2047, S2048, S2049, S2050, S2051, S2052, S2053, S2054, S2055, S2