player-31jan05-ckiger
 player-03feb05-weathers
 player-03feb05-ntroy
 player-03feb05-ckeharvey
 player-03feb05-culanga
 player-03feb05-csu
 player-01feb05-mzhang2
 player-01feb05-gliang
 player-01feb05-gbaone
 controller-31jan05-ksubman1
 cardgen-31jan05-ndennis
 banker-sk-01feb05-robertg

BlackJack Smart Player, Kiger

- Replica of Basic Player
- Stays with anything above 18: 19,20, 21
- Hits for anything below 13: 12 and lower
- When 13 thru 18, look at Dealers shown card, if value is greater than dealer value plus 10 then stay, otherwise hit
- Betting
 - Start with bet of one, if lose the double to 2, if win then bet one again
 - Every time player loses, bet is doubled, once the player wins, the bet returns to 1 and starts over again

ECE 552 In-Class Project Blackjack

Brandon Weathers

2.1.05



ECE 552 In Class Project – 1

Overview

Objective

- Given last year's VHDL files transform basic player module into "smart" player
- Smart players will compete in series of 100 hands played to determine winner
- Must abide by standard blackjack rules

Guidelines

- Bets are made before player sees cards and can not be changed
- Player can only play one hand per game
- Player with most chips after 100 hands wins

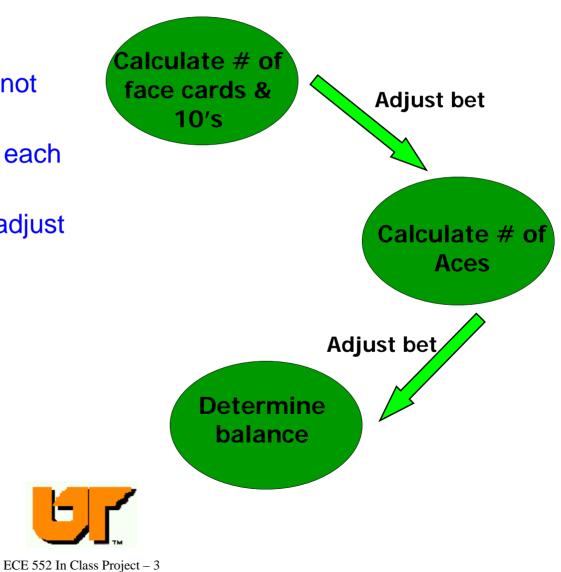


ECE 552 In Class Project – 2

Smart Player Strategy

Betting Strategy

- Splits and double downs not possible
- Must adjust bet made for each hand
- Create state machine to adjust bet for
 - Possibility of blackjack
 - Number of chips left



Smart Player Strategy

Standard Blackjack Playing Strategy



ECE 552 In Class Project – 4

6 S	Dealer's Face Up Card													
		2	3	4	5	6	7	8	9	10	A			
	8 or less	H	H	н	H	H	H	н	н	н	H			
	9	H	D	D	D	D	H.	H	H	H	H			
	10	D	D	D	D	D	D	D	D	н	Н			
	11	D	D	D	D	D	D	D	D	D	н			
	12	H	H	S	S	S	H	H	н	н	н			
	13	\$	S	S	S.	\$	н	н	н	н	н			
Υ	14	S	S	S	S	S	н	H.	н	H	н			
	15	S	S	S	S	S	H	H	H	н	н			
0	16	\$	s	S	S	S.	H	н	н	H	H			
	17 plus	S	S	S	5	S	S	S	5	S	S			
u	A9-10	S	S	S.	S.	S	S	S.	5	S.	\$			
	A-8	S	5	S	S	S.	5	S	S	S.	\$			
r	A-7	S.	Ds	Ds	Ds	Ds	5	S	н	н	н			
	A-6	н	D	D	D	D	н	H	н	н	н			
	A-5	H	н	D	D	D	н	H	н	H	н			
н	A-4	н	H	D	D	D	H	H	н	н	н			
	A-3	н	H	H	D	D	H	H	н	н	H			
a	A-2	H	H	н	D	D	H.	H	H	H	H			
1000	A-A	5.P	5P	SP	54	sP	51	SP	sP	sP	SP			
n	10-10	S	\$	\$	S.	S	S.	5	S	S	\$			
d	9-9	SF.	t, ₽	SP	sP	9P	\$	SF	sP	S.	S.			
	8-8	SP	sР	БP	SP	SP	÷Ρ	SP	SP	S.P	SP			
	7-7	ЪF	542	ъP	ΕP	S₽	SP	H	H	н	н			
	6-6	SP	SP	sP	SP	SF	H	H	н	н	н			
	5-5	D	D	D	D	D	D	D	D	H	H			
	4-4	H	н	H	ЬP	SP	H	Н	H	H	H			
	3-3,2-2	S-P	SP	SP	Þ.f	SP	SF	н	н	н	н			
3 8	3	2	3	4	5	6	7	8	9	10	A			
1.1	Dealer's Face Up Card													
Hit Stand Double/Hit Ds Double/Stand Hit														

ECE 552 Blackjack Player

• Neil Troy

Player States

- 0000 Betting Time
 - Set bet
- 0001 Look at 1st Card
 - Check if Ace, track total (assumes Ace is High)
- 0010 Look at 2nd Card
 - Check if Ace, track total (assumes Ace if High)
- 0011 Play Time
 - Check Total (try to maximize if aces are present)
 - Applies logic to determine hit/stand

Player States contd.

• 0100 Settling the Bet

- Read balance
- Next state is 0000

Hitting

- If total > 17 Stand
- If total > 13 and dealerShow < 7 Stand
- Else Hit

Suggestions

- Add code to track the game during off time
 - May require rewriting of the states
- Cumulative Probability of Different Sums
 - Answer the question how many cards are there less than x?
- Random Number Generator
 - So all players can 'take a chance'
 - Find way to use /dev/random
 - echo "obase=2; ibase=10; `head -n1 /dev/random | od -D | awk '{print
 \$3}' | head -n1`" | bc | cut -c1-16 > seed.txt

Smart Player

Karen Harvey

How Smart Player Works

- Uses card counting to determine if player should hit or stand
- If total is less than 16 always hit
- If total is 20 then stand

If total is 16 or greater then check card status to decide to hit

How Smart Player Works

Check the number of cards left that would not cause a total over 21

Check the total number of cards left to be dealt

If these two numbers are within a certain range then get another card

BlackJack - The Smart Player

Chandradevi U

Current system:

Process driven by the reset and clock signals.

Depending on the mode/state of the game, the player reacts by setting signals of hit, stand and bet.

States/Mode:

- State "0000" : Start of a game, prompts the player to set a bet. The next state is then set.
- State "0001" : If a card was issued, the total value of the card is updated and the next state is set.
- State "0010" : The state when a second card was issued. The total is updated and next state is set.

State "0011" : State to decide if the player wants to stand or hit.

- If the total > 16
 - If total > 21 and an ace was issued, the total is corrected for the ace value to be 1 and a loop to the same state is spun.
 - In all other cases, when total > 16, the player stands and the state is set to next state where the balance money is distributed to the players.
- Else if the total < 16</p>
 - And if a card was issued, the total is updated and the player remains in the same state to decide his next move.
 - If a card was not issued, the hit signal is set and the player waits for a card to be issued.

State "0100" : State to receive the money when the balance is ready and to set the state for start of the game.

Enhancements

- In State "0011", when the total is > 16 but less than 21, depending on the cards available in the deck, the player can decide to hit rather than to 'stand' always.
- After the balance is distributed, a % won can be calculated and depending on a threshold value the player can decide to stay on in the game or go to silent/watch mode for a game or to quit!

Probable Errors :

The bet amount is always set to "1" in basicPlayer when "0000" => -- Bet Mode myBet := "0000000001"; -- use smarts to set this bet <= myBet;</p>

The ace for the house player is counted as "1010".

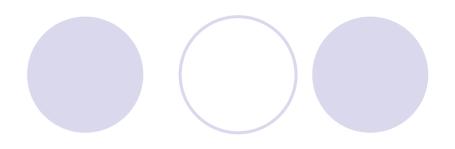
The stand flag is also set when the bet is decided.

Thank you.

Black Jack Simulation

	_						wave	– defau	ult										-
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>I</u> nsert F <u>o</u>	ormat	Tools	Windo	W															
🔁 🖬 🖨 👗 🖻 🛍 🗛 📐	<u>∦ 1</u> ± :	•	L (e 🙆 📴	24		1 II X												
		1			1) 						h dih dih dih a dih a dih					1 17
																	100101 -1 100		
— Controller ———																			
/top_tb/clock	0		u 													<u></u>		000000015	
<u> </u>	5		0.00000.000															<u>5</u>	
⊡-∏ /top_tb/dealershow	2										<u>, () (6) () (1</u>				70107	codo		11112	
⊡-☐ /top_tb/card	11		• • • • • • • •	, () , () , () () () ()	8 818 818					8 () 8 8 8 8 8 1		1 (1 0 (1 - 1 - 1		<u>, </u>		I () () () () ()	() () () ()		
ば /top_tb/cgissue ば /top_tb/issued	0	┿┿┿╋			╉╫╫╉		╋╋		┼┼┼┼┼				┼┼┼┼		₩₩				
— Players — — — — — — — — — — — — — — — — — — —	V 																		
⊡- <u>⊐</u> /top_tb/controll		10101	1 111	1	10010	0 11 0	1	101101		11111	- I I - I - II	1110	00101	111	≬I IIII	111.11	1001	1111I <mark>(</mark>)	
	000		••••••					<u></u>						• • • • • • •				• • • • •	
⊡-∏ /top_tb/pstand ⊡- <mark>∏</mark> /top_tb/bet	000 Z																	<u> </u>	
— Banker/SK — — —	۲ 	11111												****					
⊡- <u>≓</u> /top_tb/balance	127	1 0100		11101	10 H I 🕸	1111	101 10	ų III	1 11 11		111	1 10	00.011	1 11	11 11	1110	10011	1100	
📕 /top_tb/balancerdy	0			шш				цшц			ЩШЦ	ЦШП			ЩШ	ШЦ		ЩШ	
/top_tb/skbust	0		┉┉			****					*****								
<pre></pre>	00000	<u> </u>	<u>╷╷╷╶</u> ┛┈╷ ┫║┨║║║┨┃	▖▙▟▖⋏▁▁⋏⋏ ▎▌▐▏▌▋▋▋▋	▙ <u>ं</u> ⋏▙ ▋▋▋▋▋	·····································	<u>╷╷╷╷</u> ╻╷╷				┺ᄊ┻んん ┛ ║║║║┃┃┃					·····	┈╅ _╋ ん⋏ ┃┃┃┃┃	└──⋏──⋏⋏── ▎║┃║┃║	
<pre>/top_tb/handover</pre>	0																		
— Statistics ———				10					100										
⊡- <mark>∏</mark> /top_tb/numa ⊡- <mark>∏</mark> /top_tb/num2	1																	<u> </u>	
<pre></pre>	1	ul <mark>unuluu</mark> Y T Yo																<u> </u>	
⊡_ <mark>∏</mark> /top_tb/num4	0	No No										0000							
⊞- <mark>∏</mark> /top_tb/num5	0	°		<u>)((</u>													<u>(()0)((0</u>		
⊡ <mark>-</mark> _ /top_tb/num6	0																		
⊡- <u>⊐</u> /top_tb/num7 ⊡- <u>⊐</u> /top_tb/num8	1		<u>, , , , , , , , , , , , , , , , , , , </u>			<u>ou no</u> mio n				10 10 110 10 10 1 10		<u>110</u> 1110 mm							
⊡_ <mark>_</mark> /top_tb/num9				<u>, 10 10 10 10 10 10 10 10 10 10 10 10 10 </u>															
Now) ns					uluuu Dius) us		ninnn		111 120 us		mili			us
Cursor 1		2611 r	IS			J US			0	J US				120 45				100	-45
		\triangleleft																	
0 ns to 168 us	Now:	160 us I	Delta: 2																
									1										

Card Generator



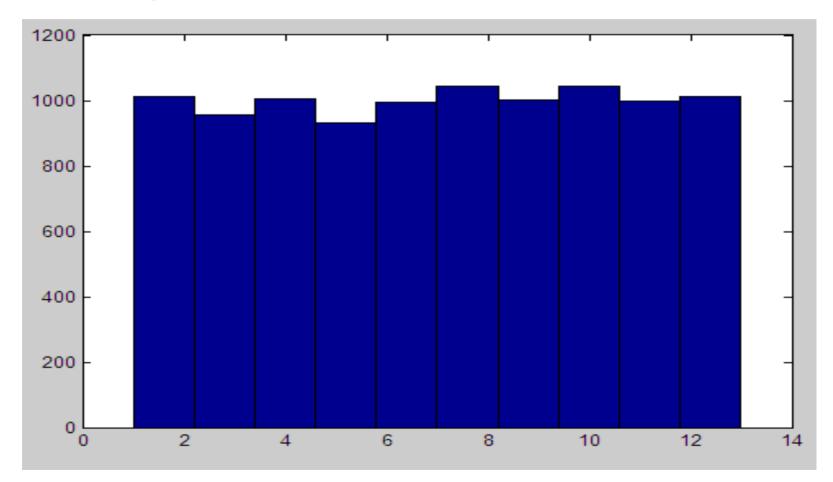
- The previous cardgenerator.vhd uses LFSR (Linear Feedback Shift Register) to generate pseudo random numbers
- Uniform random variable can be used to replace LFSR
- The algorithm used to generate Uniform random variables is Linear Congruential Method (LCM)
 a is a constant mutiplier
 c is called the increment
 m is the modulus

$$X_{i+1} = (aX_i + c) \mod m$$

 $i = 1, 2, 3, 4..$

Matlab

Histogram of Uniform Distribution



Monte Carlo Simulation

- Monte Carlo simulation techniques may be applied in the cardgenerator.vhd
- Monte Carlo simulation, which randomly generates values for uncertain variables over and over to simulate a model.
- Normal RV, Uniform RV, Exponential RV or other type RV may be used to model card generation.

Player- Basic Strategy

- Stay, when the value of card greater than 18.
- Hit, when the value of card less than 18 or less than house player.
- Increase the number of chips when I am winning
- Decrease the number of chips when I am losing

ECE552 Project "Backjack" Report1

Student: Mo Zhang Role: Player Date: Feb 1, 2005

- 01. 4 decks of cards are used for this game.
- 02. Play 100 hands or until all players are out.
- 03. In all the games the player competes only against the dealer, not other players.
- 04. At first each player starts with 50 chips and House player with unlimited chips.

- 05. The "card-generator" generates fixed sequence or random sequence of cards. The "controller" controls the sequence generation.
- 06. All cards are dealt face up for the players.
- 07. P1 (player at position 1) is always given the first card (face up) and then all other players and the house player receive the first card (face up). Then second card is given just like the first card except the house player receives 2nd card face down.

- 08. The goal of this game is to draw cards until your hand adds up to 21, or comes as close as possible without going over 21 (called "bust"). An ace can either be counted as 11 or 1, which depends on the players wish.
- 09. "card-counter" displays the compressed list of remaining cards (also the last n number of cards played).
- 10. Players should bet 1 chip to play a hand.

- 11. P1 starts playing first asking for "hit" or "stand" and then followed by all other players. A player may "stand"
 -- play just the two cards originally dealt or may "Hit"-take another card. After being dealt an additional card, the player may stop or may take still another card. A player may take as many cards as he or she wants, but as soon as the player's total exceeds 21, the player loses.
- 12. Any player who got a total of 21 with his two cards given, he is a "BLACKJACK" and he is not allowed to play that hand (since he already won).

- 13. After all players have drawn, the house player's remaining card is exposed. Then house player asks for "hit" or "stand" and according to the final result, the "Score-Keeper" decides the scores (win, loss).
- 14. If house player "busts" by going over 21, any players still in the game win. Otherwise, players with totals higher than the dealer win, while players with totals less than the dealer lose. In case of a tie, or "push" the player's bet is returned.

- 15. Player who won gets 2 chips, "blackjack" gets 3 chips, and "push" gets 1 chip.
- I6. Scorekeeper keeps the statistics of the wins and the losses of each player and finally it calculates rank of each player.

Vhdl file Explanation

Player1 <u>basicPlayer.vhd</u> bet=1
Player2 <u>fileBasicPlayer.vhd</u> bet=3
Player4 <u>housePlayer.vhd</u> bet=1023

Vhdl file Explanation

basicPlayer.vhd

- (line 3) case state is
- (line 4) when "0000" => -- Bet Mode
- (line 14) when "0001" => -- get Card 1
- (line 38) when "0010" => -- get Card 2
- (line 121)
- when "0011" => -- Game Interaction (adjusting...)
- when "0100" => (line 180)
 - -- Settle me up

Vhdl file Explanation

basicPlayer.vhd

(line 97) ----"state 0001 or 0010" if card = "0001" then -- if ace ace := ace+1; -- increment number of aces total := total + "1011"; (total=total+11)

```
(line 125) ----"state 0011(adjusting)"
    if total > "010000" then
    if total > "010101" and ace > 0 then -- not a bust? (total>21 & ace>0)
    -- account for aces
    ace := ace - 1;
    total := total - "1010"; (total=total-10)
```

bet <= (OTHERS => 'Z'); --set the bus, and every digit of "bet", to be high impedance

Example of results

playerOut.txt

(line 1) receiving card1 = 1 receiving card2 = 2 initial total = 13

hitting on 13, card received = 11 total = 23 adjusting total for Ace total = 13 hitting on 13, card received = 4 total = 17 standing on 17

11

Example of results

playerOut.txt

```
(line 485)
receiving card1 = 11 (total=total+10)
receiving card2 = 1 (total=total+11)
initial total = 21 (10+11=21)
standing on 21
```

Simulation 0-0.8us

				wave –	derault				
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>I</u> nsert F <u>o</u> rmat	<u>T</u> ools <u>V</u>	<u>N</u> indow							
a 🖬 a 💧 🗴 🏷 🖄 🖄 🕹 👗 🖄 🗎	: 🛨 🛛 🖹	L . C	🔍 🚮 😽		Bł 🕅				
🖅 /top_tb/basicplayer_1/state	0000	0000	X0001			X0010	X00	11	
<u>→</u> /top_tb/basicplayer_2/state	0000	0000		X0001			10010	X0011	
🔀 /top_tb/clock	0		\vdash \sqcap \sqcap						
<u>→</u> /top_tb/controller_1/state	5		I1						
- Card Generator								<u>,=</u>	
⊕- <mark>∭</mark> /top_tb/dealershow	2	1	Х9						
<u> → → → → → → → → → → → → → → → → → → →</u>	11	10			<u> </u>	X1	χ3	X2	<u> </u>
/top_tb/cgissue	0								
💥 /top_tb/issued	0								
— Players —	_								
⊕_∭ /top_tb/controller_1/playerid	0	0	11 X2	<u>1</u> 4	<u></u> Х1	12	<u>1 χ</u> 2	<u></u> Х1	
⊡-∭ /top_tb/phit	000	000							001
⊕-₩ /top_tb/pstand	000	1000	X001 X00	0 %010 %000	100 X000				
⊞- <mark>∭</mark> /top_tb/bet	z		(1)	(3)	(1023)				
— Banker/SK —									
⊡- <mark>∭</mark> /top_tb/balance	127	X	(127						
/top_tb/balancerdy	0								
💆 /top_tb/skbust	0								
⊕-y /top_tb/action	000000	000000							
/top_tb/takeaction	0								
🙀 /top_tb/handover	0								
Statistics									
⊡- <mark>∭</mark> /top_tb/numa	1	10				<u> </u>			
<mark>⊞-∭</mark> /top_tb/num2	0	(0						X1	
⊡-∭ /top_tb/num3	1	10					χ <u>1</u>		
⊡ <mark>_∭</mark> /top_tb/num4	0	0							
⊡ <mark>-∭</mark> /top_tb/num5	0	10							
/top_tb/num6	0	10							
<mark>⊡-∭</mark> /top_tb/num7	1	(0							
<mark>⊞-∭</mark> /top_tb/num8	1	(0							X

Simulation 0.8-1us

			wave –	default						r
ols <u>v</u>	<u>w</u> indow									
	<u>.</u> 19. 19. 9. 9.	🔍 👫 34	27 EL C)	E4 🐹						
000	0000	(0001			(0010	<u>(00</u>	11		<u>(0100</u>	
000	0000		X0001			0010	(0011			
		<u>)(</u> 9								
	<u>(0</u>			<u>,12</u>			_ <u>`2</u>	<u>,8</u>		<u>,11</u>
	<u>XO</u>	(1)(2)(1		1 /2			<u>}2</u>	
)0	000							<u>,001 (00</u>		<u>(010) (</u>
)0	.000	<u> </u>	<u>0 (010 (000)</u> (3)	<u>100 (000</u> 1023)					<u> X001 X000</u>	
27	X	(127								
0000	υυυυυ									
	(0				X1					
					<u>^1</u>		(1			
	<u>10</u>					<u>)</u> 1				
	<u>.</u> (0									
	.o									

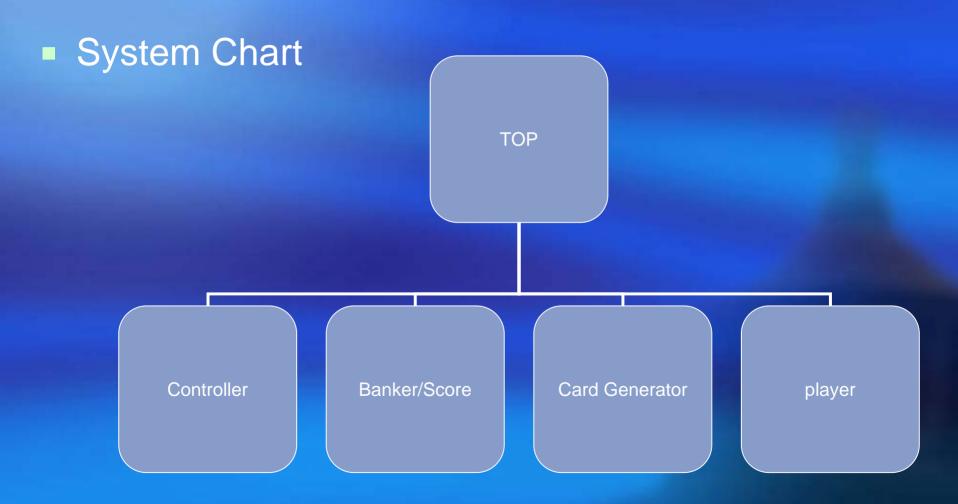
Simulation 1-1.8us

() @ ¬= «= ma) =% /% Ľ	·				C				
p_tb/basicplayer_1/state	0000	0100					(0000		
p_tb/basicplayer_2/state	0000	0011	<u>,01</u>	00				(0000	
oller ————									
p_tb/clock	0								
p_tb/controller_1/state	5	3		χ6		<u>)</u> 4			<u>,5),0),1</u>
Generator ————									
p_tb/dealershow	2	9							
p_tb/card	11	11	4		11				
p_tb/cgissue	0								
p_tb/issued	0								
`s									
p_tb/controller_1/playerid	0	2		X4		X1	χ2	<u>)</u> 4	1 0 (1
p_tb/phit	000	000 (010	(000	(100	<u>(000</u>				
p_tb/pstand	000	000		0 X000	, ,10	0 X000			
p_tb/bet	z								
/SK									
p_tb/balance	127	127					(126 (13	0 (127	
p_tb/balancerdy	0								
p_tb/skbust	0								
p_tb/action	000000	υυυυυυ				(0010	00 X000000		
p_tb/takeaction	0								
p_tb/handover	0								
stics									
p_tb/numa	1	1							<u>xo</u>
p_tb/num2	0	1							Xo
p_tb/num3	1	1							Xo
p_tb/num4	0	0	11						χ <u>ο</u>
p_tb/num5	0	0							
p_tb/num6	0	0							
p_tb/num7	1	0							
p_tb/num8	1	1							X0
p_tb/num9	0	 0							
p_tb/num10	0	0							
p_tb/numj	2	1			2				<u> </u>

Black Jack

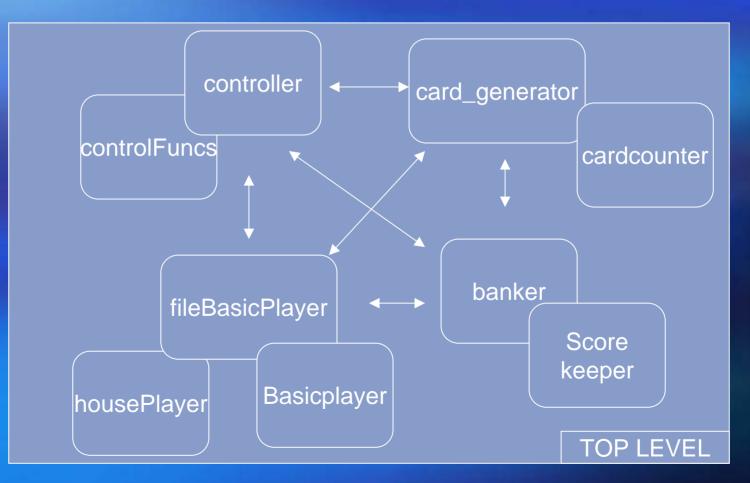
Improvement of PLAYER group ECE 552 Spring 2005

Overview of BJ



Submodule

Divided into 9 submodule



Simulation

1. Using the BJ.do file to do the simulation, and run for 165000ns.

2. Two basic player and one house player was in game.

— Player States — — — — — — — — — — — — — — — — — — —	
	, 2004) min sonands found for a number of the for a min subscription in no second in <mark>t</mark> he for a second statement in submit a submit and the second
⊡-∭ /top_tb/b 0000	na – 14 – 1 – 1 – 1 – 1 – 1 – 1 – 1 – 1 –
- Controller	
📕 /top_tb/c1	
⊡-∭ /top_tb/c4	a ale al lianti manti manti callationati na al la matana nationa matali a <mark>ta calina na la stati sati a atti a anna</mark>
🞽 /top_tb/g0	
/top_tb/g0	
— Card Generator	
I /top_tb/d1	
/top_tb/c0	
/top_tb/i0	
- Players	
Hop_tb/phit 000 Top_th/phit 0000 Top_th/phit 0000 Top_th/phit 0000 Top_th/phit 0000	<u> </u>
<u> </u>	իկի պատուստական ակիկի սոտվակվու իրանիությունը ավերակությունը ակոտովերիու <mark>ն</mark> ու պատովետիրության կանիության ակիստոն ա
	<u>, , , , , , , , , , , , , , , , , , , </u>
— Banker/SK —	*
/top_tb/b0	
/top_tb/s0	
/top_tb/t1	
/top_tb/h1	
- Statistics	
— statistics ⊞−7 /top_tb/numa 0	
Hard Top_tb/num2 1	
\blacksquare /top_tb/num41	
\blacksquare /top_tb/num5 0	
H= / /top_tb/num6 0	
$H = 1/10p_10/num/0$	
H= / /top_tb/n0	
/top_tb/eU	
/top_tb/r0	
/top_tb/bU	
/top_tb/g0	
/top_tb/c0	
/top_tb/c0	
Now 165000 ns	0 4 us 40 us 120 us 160 us
Cursor 1 103700 ns	

Basic function

- Two player and dealer
- Fixed value for each player
- Cardgenerator try to use uniform distribution to generate each card
- Use 4 new decks for each hand
- Dealer obey the soft 17 rule

Possible Improvement

Maybe more effective way to make card generator.Add more actions to player for more fun

Player's action in a hand

— Player States ————										
<mark>⊞-∭</mark> /top_tb/basicplayer_1 00	11 0000	0001	(0010	(0011		(0100			0000	00
<mark>⊞-∭</mark> /top_tb/basicplayer_2 00	11 0000	(0001		010 (00	11	(01	00		(00	00
— Controller —————										
🗾 /top_tb/clock 🔰 1		JUUUU	JUUU	JUUUL	mm	MUUU	mm	mm	JUUUU	
□-j /top_tb/controller_1/ 3	4 (5)(0)	1	<u> 1</u> 2	(3),e	5	(4		<u>)(5)(0)(1</u>
🔰 /top_tb/gameover 🛛 🛛 🛛				149. 						
🗾 🔰 /top_tb/globalreset 🛛 🛛 🛛									·	
— Card Generator —————										
Image: Image	9) <u>(</u> 3								<u></u> 1
<mark>⊞-∭</mark> /top_tb/card 12	11		3 (13	(1) (8	(13)	3	<u> </u> 2	(10		
🗾 📶 /top_tb/cgissue 🛛 🛛 🛛		<u> </u>								
📕 🗾 /top_tb/issued 🛛 🛛 🛛 🖉										
— Players —————————				e da dita	2	9 67 68		57 - 56a - 1		
- definition and the set of the	<u>4)</u> 0 (1 (2 (4	<u>1 2 1</u>	2 (1)2 (4		(1	2 (4	XO X1 X2
<mark>⊞–</mark> ⊒/top_tb/phit 00	0 000			<u> </u> χ		1000		000		
B→J /top_tb/pstand 00	0 000		(000				00)()(00)	
<mark>⊞-∭</mark> /top_tb/bet Z		(1)-(3)-(}							(1)
— Banker/SK —————	1000			and but we						
<mark>⊞-∭</mark> /top_tb/balance 13	6 127			1 1 1	26	X	(127	L I		27)
🔰 📈 /top_tb/balancerdy 🛛 0										
🛒 /top_tb/skbust 🔰 0									1	
<u>-</u> <u>-</u> <u>-</u> <u>/</u> top_tb/action 00	00 00000	0						X	(00000)	þ
🗾 /top_tb/takeaction 🛛 0	1									
📕 📈 /top_tb/handover 🛛 🛛 🛛 🖉										
- Statistics								1.55		

 $\begin{array}{c} \text{Bet} & & \text{Get 1st card} & & \text{Get 2nd card} \\ \hline \\ & & & & & & & \\ & & & & & \\ & & & & & & \\ & & & &$

Basic function for player

- Fixed bet value
- Only STAND and HIT available
- Simple strategy based on the same as dealer – hit until total greater than 16
- Actions independent from dealer's card and former hands

Goal

Try to find a better strategy depend on the show card
Maybe bet intelligently based on the previous result



BLACKJACK GAME ECE 552

THE PLAYER

Gaurav

BLACKJACK PLAYERS HOUSE PLAYER - DEALER -HAS UNLIMITED CHIPS (1024) -FIRST CARD DEALT IMMEDIATELY AFTER RESET = '0'- SECOND CARD DEALT ON HIS TURN -BET PLACED - ALWAYS '0' CHIPS - COMPETES WITH ALL THE PLAYERS

BLACKJACK PLAYERS CONTD...

BASIC PLAYER

HAS LIMITED CHIPS (128)
FIRST AND SECOND CARDS DEALT IN CONSECUTIVE PLAYER MODES
ABILITY TO "*HIT*" OR "*STAND*" ACCORDING TO PLAYER STRATEGY
BET PLACED – ALWAYS 1 CHIP
COMPETES ONLY WITH THE DEALER

PLAYER STATES

• STATE 1: <u>BET MODE</u>

- -state="0000"
- House player bets 0
- Basic player bets 1 chip. (myBet <= "000000001").
- Both players have stand signal high

	0000	0000) 0001			20010	00	11	i se se s	0100
	0000	0000		0001			0010	X0011		
Controller										
📕 /top_tb/clock	1									
	1	0	1		1/2			/3		
Card Generator										
⊕ /top_tb/dealershow	1	X	/9							
⊞–_ /top_tb/card	0	ω			<u>(12</u>	χ1	/3	12	/8	
🗾 /top_tb/cgissue	U									
🗾 /top_tb/issued	0									
Players										
──── /top_tb/controller_1/playerid	1	Δ	1 /2	X4	<u>Х1</u>	1/2	<u>11 /2</u>	X1)
── <mark>─</mark> /top_tb/phit	000	000							1001 Xooc	0
── <mark>─</mark> /top_tb/pstand	000	000	<u>/001 /00</u>	0 /010 /000	100 1000					X001 X
⊞– <mark>,</mark> /top_tb/bet	Z		()	-3	(1023)					
Banker/SK										
	X	Х	(127							
🗾 /top_tb/balancerdy	U									
🗾 /top_tb/skbust	U									
	υυυυυυ	UUUUUU								
📕 /top_tb/takeaction	U									
🗾 /top_tb/handover	0									
Statistics										
⊡— /top_tb/numa	0	ю				<u> </u>				
⊞- <mark></mark> /top_tb/num2	0	ω						X1		
⊡—_ <mark>_</mark> /top_tb/num3	0	ω					<u>Х1</u>			
⊞- <mark></mark> /top_tb/num4	0	ω								
	0	Δ								
⊞— <mark>, /</mark> top_tb/hum6	0	Δ								
	0	0								
	0	۵ ا							X1	
. ⊡ /top_tb/num9	0	Δ								
⊕–_ /top_tb/num10	0	۵ ا								
	0	۵ ا								
Now	640000 ns		0 20		00 40		00 60			
Cursor 1	100 ns	100								

PLAYER STATES contd...

• STATE 2: <u>CARD 1 DEALT</u>

- state="0001"
- If card is issued on the bus update card total.
 - If card is an ace, total = 11 and number of aces = 1
 - Else, total = total + card (card < 10)
 - Or total = total + 10 (card = 10/ J/ K/ Q)
- In case of *House Player*
 - Total = dealershow
 - *dealershow* signal from controller displays dealer's card value.

/top_tb/basicplayer_1/state	0001	0001			0010) 00	11		0100	
⊕ /top_tb/basicplayer_2/state	0001	0000 /0001				0010	0011			
Controller										
🗾 /top_tb/clock	1									
⊕ /top_tb/controller_1/state	2	1	12)3			
Card Generator										
	9	9								
/top_tb/card	12	0		12	<u></u>	/з	/2)8		X11
🗾 /top_tb/cgissue	1			1						
🗾 /top_tb/issued	1			l T						
Players										
⊕ /top_tb/controller_1 /playerid	1)2)4	<u></u>)2)	1 /2	X1		/2	
· ⊞– /top_tb/phit	000	000						001 000		010 X000
/ ⊕– /top_tb/pstand	000	1000 1010 1000)001)000	
⊞– <mark>,</mark> /top_tb/bet	Z	p⊣₅	1023	+						
Banker/SK										
	127	127								
🗾 /top_tb/balancerdy	0			_						
🗾 /top_tb/skbust	0									
⊕-☐ /top_tb/action	υυυυυυ	υυψυυυ								
🗾 /top_tb/takeaction	U									
🗾 /top_tb/handover	0									
Statistics										
	0	0			X1					
	0	0								
	0	0				X1				
	0	0								
	0	0								
	0	0								
	0	0								
	0	0						X1		
	0	0								
E—_ /top_tb/num10	0	0								
	0	0								χ1
Now	640000 ns	200 30		1 40	0 50	0 60	0 7(0 80	0 90)0 1us
Cursor 1	380 ns) ns						

Player 1 = 12

Player 2 = 3

Dealer = 9

PLAYER STATES contd...

• STATE 3: <u>CARD 2 DEALT</u>

- state="0010"
- For Players it is same as STATE 2.
- In case of *House Player*
 - If card is issued on the bus update card total.
 - If card is an ace and number of aces = 1 then increment total by 1 else increment total by 11 and increment number of aces by 1.
 - Else, total = total + card (card < 10)
 - Or total = total + 10 (card = 10/ J/ K/ Q)

	0010	0010		\	11) 0100			
	0010	0001	0010		X0011					Xo100
Controller										
🗾 /top_tb/clock	1									
<u>-</u> <u>-</u> <u>/top_tb/controller_1/state</u>	2	2			/3					/6
Card Generator										
	9	9								
	3	12 (1		3	2)8		X11		4
📕 /top_tb/cgissue	1									
🗾 /top_tb/issued	1									
Players										
	1	2	Ϊ <u>ι</u>	2_			12			<u>14</u>
⊞– <mark>,</mark> /top_tb/phit	000	000				Xoo1 Xoo		X010 X00	0 X010	X000 X100
✓ ⊕ /top_tb/pstand	000	000					2001 2000			X010 X000
	Z									
Banker/SK										
	127	127								
🗾 /top_tb/balancerdy	0			_						
📕 /top_tb/skbust	0									
	υυυυυυ	000000		_						
/top_tb/takeaction	U									
📕 /top_tb/handover	0									
Statistics										
	1	<u>0 X1</u>								
	0	0								
	1	0		1						
	0	0								1
⊡— <mark>—</mark> /top_tb/num5	0	0								
	0	0								
	0	0								
	0	0				X1				
	0	0								
/top_tb/num10	0	0								
⊡— <u> </u>		0						X1		
	Now 640000 ns	5	l 00	LIIII 60		1 i i i l i i i i 00 80	liiiiliii 00	1 i i i l i i i i 300 1	us 11	00 1200 13
Cur	rsor 1 544 ns			ns						

Player 1 = 8

Player 2 = 11

Dealer = 11

PLAYER STATES contd...

• STATE 4: <u>GAME INTERACTION</u>

- -state="0011"
- Player Strategy
 - If total is above 16
 - If total is above 21 adjust the number of aces and make ace value = 1
 - Else stand
 - If total is not above 16, hit and increment total until total is above 16.

		ta	0100	0011			0100						
	→ / top_tb/basicplayer_2/stat		0011	0011		≓	0100			<u>ل</u> مر	00		
	— Controller	16				+				/01	00		
	/top_tb/clock		1		\vdash \frown \leftarrow					\square \square \square		\square \square \square	
			3	2)3		H							
	- Card Generator		Ŭ	<u> </u>		T							
	⊕ /top_tb/dealershow		9	a									
	⊕– <mark>,</mark> /top_tb/card		8	2	18	F		X1 1		4		li 1	
	/top_tb/cgissue		0	Fi -		Ħ		,					
	/top_tb/issued					+							
1			-										
		erid	1	2/1			2) <u>4</u>		X1
	·····································		000	000	1001 Xoo	00		010 /000	0 /010		X100	1000	
			001	000			001 \000				0 /000) /000
	⊕– <mark>, /top_tb/bet</mark>		Z										
-	Banker/SK												
	⊞— <mark>, /top_tb/</mark> balance		127	127									
	🗾 /top_tb/balancerdy		0										
	🗾 /top_tb/skbust		1										
	⊞— <mark>—</mark> /top_tb/action		υυυυυυ	υυυυυ	U								001000
	🗾 /top_tb/takeaction		U										
	🗾 /top_tb/handover		0										
	Statistics												
	⊕ /top_tb/numa		1	1									
	⊕– <mark>, /t</mark> op_tb/num2		1	1									
_	⊕– <mark>,</mark> /top_tb/num3		1	1									
	⊕– <mark>,</mark> /top_tb/num4		0	0						1			
	<u> </u>			0									
	<u> </u>			0									
	<u> </u>			0									
				0	<u> </u>								
	⊞– <mark>, /</mark> top_tb/num9			0									
- 18	<u> </u>			0				V				V	
	⊞– <mark>, /t</mark> op_tb/numi			0								2	
		Now	640000 ns	1 I I I I 7(1 I I I I I I I I I I I I I I I I I I I	1 i 300	90 90	11111111 10 1	liiiliii us 11				
1		Cursor 1	820 ns			820				12	10		

Player 1 hits and gets card = 8

Then deactivates hit signal and stands.

PLAYER STATES contd...

• STATE 5: <u>END OF HAND</u>

- -state="0100"
- -Player
 - If balance is ready (balanceRdy from Banker) Player updates his balance.
 - Else waits for the balanceRdy signal to go high.
 - Player in has Stand signal ON.
- Similar for House Player

PLAYER STATES contd...

• STATE 6: <u>IDLE</u> – Player

• All shared buses on high impedance.

Some Suggestions

- Logic Modification
 - The second card to the House Player should be dealt along with the other players for a fair game. (rather than dealing it after the players play their hand)
 - Suggestion: To hide the card, the card may be generated and kept as a separate signal accessible only to the card generator and house player (may be also scorekeeper).

- House Player code errors
 - In case of an ace, adds 10 instead of 11 to the total.
 - In state 1: bet mode, bets 1024 instead of 0. (doesn't make any difference!)

THANK YOU.

BLACKJACK (Controller module)

Presented by: Karthik, Ikram, Priya & Tushti ECE 552 | Spring'05

Role of the Controller:

- •Notification of the start of a new hand
- •Cycles through players and makes them place their bets
- •Controls the assignment of cards to the players & the dealer
- Activates the Scorekeeper & the banker after all players have played

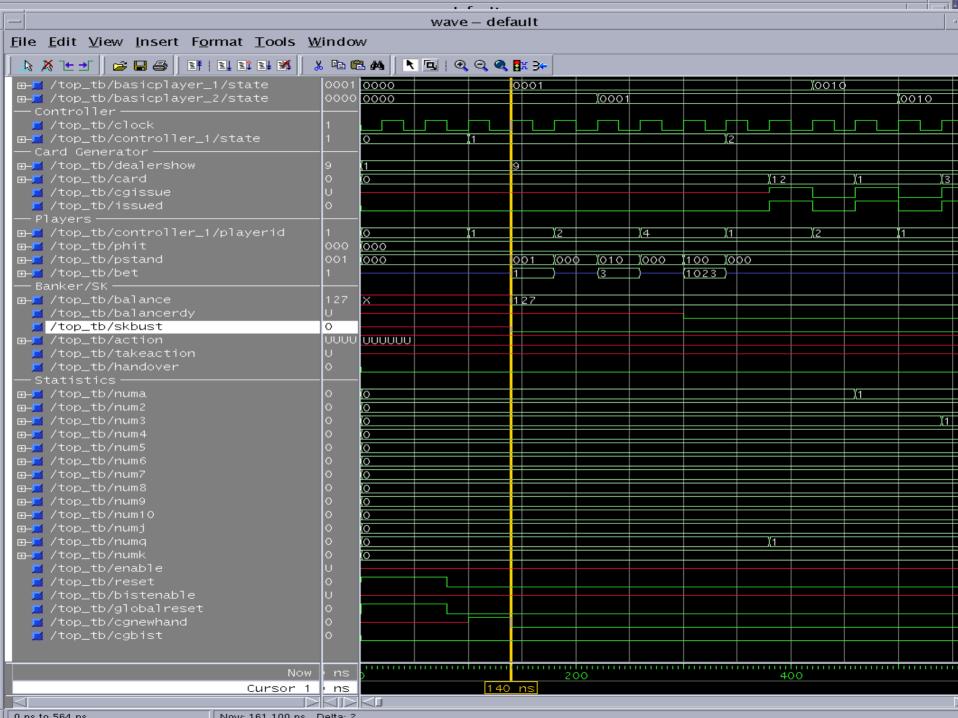
State "0000" and "0001"

State "0000"

cgNewHand <= '1' : this signifies the start of a new hand

State "0001"

All the players place their respective bets



State "0010"

All the players (excluding the dealer) get their initial 2 cards

Player 1 gets: 12,3

Player 2 gets: 1,2

Dealer gets his open card : 9

				۲	wave –	default								
<u>E</u> dit <u>V</u> iew <u>I</u> nsert F <u>o</u> r	mat <u>T</u> ools <u>W</u>	indo	w											
🔉 1± 🛨 🛛 😂 🛛 😂 🕇 📑		6 🖻	244			🔍 📴 Э	+							
/top_tb/basicplayer_)(00)	11				(0100					کا
/top_tb/basicplayer_	2/state	0000	0010			(0011								
ontroller ————														
/top_tb/clock		1												
/top_tb/controller_1	/state	1	2			(3								
ard Generator ———														
/top_tb/dealershow		9	9											
/top_tb/card		0	3			_	<u> 18</u>				<u>1</u>	1		
/top_tb/cgissue		U									<u> </u>			
tb/issued		0	P II											
tb/controller_1	/playerid	1	1	<u>)</u> 2		<u> </u>				1/2				
Netscar tb/phit		000	000				001	<u> (000</u>			<u>1010</u>	<u> </u>)	Xo
/top_tb/pstand		001	000						<u> (001</u>	<u> 1000 (</u>				
/top_tb/bet		1												
anker/SK		4.07												
/top_tb/balance		127	127											
/top_tb/balancerdy		0								<u> </u>				
/top_tb/skbust /top_tb/action			υυυυ											
/top_tb/takeaction		lu	0000	00										
/top_tb/handover		ŏ												
tatistics ———		Ľ												
/top_tb/numa		0	1								+			
/top_tb/num2		ō	ō		<u> 1</u>									
/top_tb/num3		0	1											
/top_tb/num4		0	0											
/top_tb/num5		0	0											
/top_tb/num6		0	0											
/top_tb/num7		0	0											<u>ا ا</u>
/top_tb/num8		0	0				<u>(1</u>							
/top_tb/num9		0	0											
/top_tb/num10		0	0											
/top_tb/numj		0	0								<u> </u>			
/top_tb/numq		0	1											
/top_tb/numk		0	0											
/top_tb/enable		U												
/top_tb/reset		<u>.</u>												
/top_tb/bistenable														
/top_tb/globalreset /top_tb/cgnewhand											+			
/top_tb/cgbist		0 0									+			
		Ĩ												
	Now	, ns									1			
	Cursor 1	ns ns		60	/0			80				<u> </u>	<u>45</u>	
s to 1096 ns	Now: 161,100 ns [

State "0011"

All the players (excluding the dealer) either 'hit' or 'stand'

So they get their remaining cards until they 'stand'

Final cards for:

Player1: 12,3,8

Player2 : 1,2,11,4

State "0110"

Dealing with the Dealer:

•Here the dealer gets his cards until he 'stands'

• Final cards with the dealer: 9,11

wave – default														-
<u>E</u> dit <u>V</u> iew <u>I</u> nsert F <u>o</u> rmat <u>T</u> ools <u>W</u>	indov	v												
x 1± 1 🛥 🖬 🔿 1f 1 1 1 1 1	, 🗈 🕻	3 #4 []		u 🗨	୍ର୍	, B # 34								
/top_tb/basicplayer_1/state	0001	0100										Yc	000	
/top_tb/basicplayer_2/state		0011			<u>)</u> 010	h0						^	/000	t y
Controller		0011												
/top_tb/clock	1								+ г	\dashv \square			+	
/top_tb/controller_1/state	1	3				16				4	₽ _			
ard Generator —														
/top_tb/dealershow	9	9												
/top_tb/card	0	11	4					111						
/top_tb/cgissue	υ													
/top_tb/issued	0		 					1	ī					
layers			1											
/top_tb/controller_1/playerid	1	2				14				1 1		12	2	ΤX
/top_tb/phit	000	010		Xobc)		100	X	000					ين ال
/top_tb/pstand	001	000			<u>)</u> 010	0 000			1	00 1000				
/top_tb/bet	1													
anker/SK														
/top_tb/balance	127	127										<u>1</u>	26 (13	φ
/top_tb/balancerdy	υ													
/top_tb/skbust	0													
/top_tb/action	υυυυ	UUUUU	ψυ								00100	jo Xo	00000	
/top_tb/takeaction	U													
/top_tb/handover	0													
itatistics ————														
/top_tb/numa	0	1												
/top_tb/num2	0	1												
/top_tb/num3	0	1												
/top_tb/num4	0	0	1											
/top_tb/num5	0	0												
/top_tb/num6	0	0												
/top_tb/num7	0	0									_			
/top_tb/num8	0	1									_			
/top_tb/num9	0	0												
/top_tb/num10	0	0												
/top_tb/numj	0	1						12			_			
/top_tb/numq /top_th/numq	Å													
/top_tb/numk /top_th/opshle	Ľ.	0												
/top_tb/enable /top_tb/reset														
/top_tb/heset /top_tb/bistenable	Ň I													
/top_tb/globalreset	ă I													
/top_tb/cgnewhand	ŏ													
/top_tb/cgbist	ŏ													
Now	, ns	1200			mulu		1 1 1 1 1 1 1 1 4 0				1 111111111 1600			
Cursor 1	ns n	1200					140	<u> </u>			1600			
		$\leq \Pi$												\sim

ns to 1852 ns Now: 161,100 ns Delta: 2

State "0100" and "0101"

State "0100"

Results stage: control goes to the scorekeeper to decide the winner

State "0101"

Gameover stage : Displays the final amount of chips won by each player

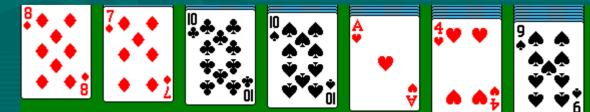
File Edit View Insert Format Loots Window Image: A construction Image: A construction <td< th=""><th></th><th>wave – default</th><th>- 🗆</th></td<>		wave – default	- 🗆
mm /top_tb/bastcpla 0000 00	<u>File Edit View Insert Fo</u>	ormat <u>T</u> ools <u>W</u> indow	
mm /top_tb/bastcpla 0000 0000 0000 mm /top_tb/lostcpla 0000 0000 0000 mm /top_tb/lostcpla 0 0 0 0 mm /top_tb/lostcpla 0 0 0 0 0 mm /top_tb/lostcpla 0 0 0 0 0 0 mm /top_tb/lostcpla 0 0 0 0 0 0 0 mm /top_tb/lostcpla 0 <td< th=""><th></th><th></th><th></th></td<>			
ms /top_tb/casicpla 0000 0000 0000 controller 000 0000 0000 ms /top_tb/controll 5 4 15 0 1 -card Generator 2 9 1 3 1 1 -card Generator 2 9 1 3 1 1 -card Generator 2 9 1 3 1			
Controller- Controller- Controller- Card Generator Card Ge			
mail /top_tb/controll 5 4 5 0 1 mail /top_tb/dalershow 2 3 3 3 mail /top_tb/card 1 1 3 3 mail /top_tb/card 0 1 3 3 mail /top_tb/card 0 1 3 3 mail /top_tb/card 0 4 0 3 3 mail /top_tb/staud 0 4 0 3 3 mail /top_tb/bal 000 00 00 00 00 00 00 00 00 00 <td< td=""><td></td><td></td><td></td></td<>			
Card Generator 2 9 3 B= / top_tb/card 11 11 3 I / top_tb/cigissue 0 3 P / top_tb/issued 0 1 P/ ayers 0 1 B= / top_tb/phit 000 000 B= / top_tb/phit 000 000 B= / top_tb/balance 127 13 B= / top_tb/balancerdy 127 130 127 I / top_tb/balancerdy 0 000 000 000 I / top_tb/hadcover 127 130 127 I / top_tb/halancerdy 0 000 0000 000000 I / top_tb/halancerdy 0 0 000 000000 I / top_tb/halacction 13 0 0 0 I / top_tb/num8 1 1 0 0 0 B= / top_tb/num8 1 1 0 0 0 0 B= / top_tb/num8 1 1 0 0 0 0 0 B= / top_tb/num8 0 0 0 0 <td></td> <td></td> <td></td>			
ms /top_tb/card 11 /top_tb/cgissue 0 11 /top_tb/cgissue 0 1 /top_tb/balance 127 10 /top_tb/sklaancerdy 127 10 /top_tb/sklaance 127 10 /top_tb/sklaance 127 10 /top_tb/nadover 0 1 /top_tb/num3 1 1 /top_tb/num4 1 1 /top_tb/num3 1 0 /t		5 4 5 0 1	
B=2 / top-tb/cgissue 11 1 1 / top-tb/lsued 0 Players 0 B=2 / top-tb/bhit 000 B=2 / top-tb/bhit 000 B=2 / top-tb/beit 000 B=2 / top-tb/beit 0 B=2 / top-tb/beit 0 B=3 / top-tb/beit 0 B=4 / top-tb/beit 2 B=5 / top-tb/beit 12 B=4 / top-tb/beit 12 B=5 / top-tb/balancerdy 12 B=4 / top-tb/shalancerdy 12 B=5 / top-tb/halancerdy 12 Cop-tb/skbust 0 Cop-tb/shalancerdy 0 Cop-tb/shalancerdy 0 Cop-tb/shalancerdy 0 Cop-tb/shalancerdy 0 Cop-tb/num3 1 1 1 B=5 / top-tb/num3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 <			
/top.tb/ssued 0 Players 0 B=//top.tb/controll 0 B=//top.tb/bstand 000 B=//top.tb/bstand 000 B=//top.tb/bstand 000 B=//top.tb/bstand 000 B=//top.tb/bstand 000 B=//top.tb/bstand 000 B=//top.tb/balancerdy 127 /top.tb/skbust 0000 D/top.tb/chalancerdy 0 /top.tb/chalancerdy 0 /top.tb/chalancerdy 0 /top.tb/action 0000 D/top.tb/num2 1 B=//top.tb/num3 1 I 1 B=//top.tb/num4 1 B=//top.tb/num5 0 B=//top.tb/num6 0 B=//top.tb/num6 0 B=//top.tb/num6 0 B=//top.tb/num3 1 B=//top.tb/num3 1 B=//top.tb/num4 0 B=//top.tb/num4 0 B=//top.tb/num4 0 B=//top.tb/num6 0 D 0			3
m /top_tb/issued 0 m+1 /top_tb/controll 00 m+1 /top_tb/phit 000 m+1 /top_tb/stand 000 m+1 /top_tb/stalance 127 m+1 127 130 m+1 127 140 m+1 120 140 m+1 120 140 m+1 120 140 m+1 120 <td></td> <td></td> <td></td>			
B=2 /top_tb/phit 0 1 B=2 /top_tb/phit 000 000 B=2 /top_tb/pstand 000 000 B=3 /top_tb/balance 2 10 B=4 /top_tb/balance 127 130 127 B=4 /top_tb/balance 127 130 127 B=4 /top_tb/balance 000 0000 0000 0000 B=4 /top_tb/balance 127 130 127 100 B=4 /top_tb/balance 0000 00000 000000 000000 B=4 /top_tb/hadover 0 0000 00000 00000 000000 Statistics 1 1 00 00 00 00 B=4 /top_tb/numa 1 1 00 00 00 00 B=4 /top_tb/num3 1 1 00 00 00 00 00 B=4 /top_tb/num3 1 1 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		o	
B-1 /top_tb/phit 000 000 B-1 /top_tb/balance 2 B-1 /top_tb/balance 127 1 /top_tb/balance 127 1 /top_tb/balance 000 1 /top_tb/balance 127 1 /top_tb/balance 000 0 /top_tb/hakeaction 000 0 /top_tb/numa 1 1 /top_tb/num3 1 1 /top_tb/num4 0 1 /top_tb/num4 0 0 /top_tb/num5 0 0 /top_tb/num6 0 0 /top_tb/num10 0 0 /top_tb/num10 0 0 /top_tb/num1 2 0 /top_tb/num3 1 0 /top_tb/num4 0 0 /top_tb/num3 0 0 /top_tb/num4 0 0 /top_tb/num4 0 0 /top_tb/num4 0 0 /top_tb/copist 0 0 /top_tb/copist 0 0 /top_tb/co			
B=1 /top_tb/pstand 000 000 000 B=1 /top_tb/bet 2 2 1 B=1 /top_tb/balancerdy 0 127 1 1 1 /top_tb/ablancerdy 0 0 0 0 1 1 /top_tb/ablancerdy 0 0 0 0 0 0 1 /top_tb/takeaction 0 0 0 0 0 0 0 Statistics 0 1 0 0 0 0 0 0 B=1 /top_tb/num3 1 1 0			
Barl /top_tb/bet Z Barl /top_tb/balance 127 130 127 127 130 127 130 127 130 127 130 127 130 127 130 127 130 127 130 127 000 0000 000000 127 000 127 000 127 000 127 000 0000 000000 127 000 127 000 127 000 127 000 127 000 127 000 127 000 127 000 127 000 127 000 127 000 127 000 1287 100 129 00 129 00 120 00 120 00 <t< td=""><td></td><td></td><td></td></t<>			
Banker/SK 127 130 127 Implify /top_tb/balancerdy 0 0000 0000000 Implify /top_tb/skbust 0 0000 0000000 Implify /top_tb/takeaction 0 0 0 0 Implify /top_tb/takeaction 0 0 0 0 0 Statistics 0 1 0 0 0 0 Implify /top_tb/num3 1 1 0 0 0 0 Implify /top_tb/num3 1 1 0 0 0 0 0 Implify /top_tb/num4 0 1 0 </td <td></td> <td></td> <td></td>			
/top_tb/balancerdy 0			
<pre>/ top_tb/skbust / top_tb/hadover statistics / top_tb/num3 / top_tb/num3 / top_tb/num4 / top_tb/num5 / top_tb/num5 / top_tb/num6 / top_tb/num7 / top_tb/num8 / top_tb/num8 / top_tb/num8 / top_tb/num9 / top_tb/num9 / top_tb/num10 g=/ / top_tb/num10 g=/ / top_tb/num4 g=/ / top_tb/num4 g=/ / top_tb/num5 g=/ / top_tb/num3 g=/ / top_tb/num3 g=/ / top_tb/num4 g=/ / top_tb/num4 g=/ / top_tb/num4 g=/ / top_tb/num5 g=/ / top_tb/num3 g=/ / top_tb/num4 g=/ / top_tb/sistenable g=/ / top_tb/cgnewhand g=/ /</pre>		127 130 127	
Image: Comparison of the control of			
/top_tb/takeaction 0 Jop_tb/handover 0 Statistics 1 Dif/top_tb/numa 1 Dif/top_tb/num2 0 Dif/top_tb/num3 1 Dif/top_tb/num4 0 Dif/top_tb/num4 0 Dif/top_tb/num4 0 Dif/top_tb/num5 0 Dif/top_tb/num6 0 Dif/top_tb/num7 0 Dif/top_tb/num8 1 Dif/top_tb/num3 1 Dif/top_tb/num6 0 Dif/top_tb/num7 1 Dif/top_tb/num8 1 Dif/top_tb/num10 0 Dif/top_tb/num4 0 Dif/top_tb/num3 1 Dif/top_tb/num3 1 Dif/top_tb/num3 2 Dif/top_tb/num4 0 Dif/top_tb/num4 0 Dif/top_tb/num3 2 Dif/top_tb/num4 0 Dif/top_tb/num4 0 Dif/top_tb/num4 0 Dif/top_tb/num4 0 Dif/top_tb/num4 0 Dif/to			
Image: Arrow of the second			
Image: /top_tb/numa 1 1 1 0 Image: /top_tb/num3 1 1 0 0 Image: /top_tb/num4 0 1 0 0 Image: /top_tb/num5 0 0 0 0 Image: /top_tb/num6 0 0 0 0 Image: /top_tb/num8 1 1 0 0 Image: /top_tb/num10 2 2 0 0 Image: /top_tb/num4 0 1 0 0 Image: /top_tb/num4 0 1 0 0 0 Image: /top_tb/num4 0 1 0 0 0 0 Image: /top_tb/num4 0 0 0 0 0 0 0 0 0 0 0 0 0 <td< td=""><td></td><td></td><td></td></td<>			
Image: /top_tb/num2 0 1 0 Image: /top_tb/num3 1 0 0 Image: /top_tb/num4 0 1 0 Image: /top_tb/num5 0 0 0 Image: /top_tb/num7 1 0 0 Image: /top_tb/num10 0 0 0 Image: /top_tb/num10 2 2 0 Image: /top_tb/num4 0 0 0 0 0 Image: /top_tb/sistenable 0 <			
Image: /top_tb/num3 1 1 1 0 Image: /top_tb/num4 0 1 0 0 Image: /top_tb/num5 0 0 0 0 Image: /top_tb/num6 0 0 0 0 Image: /top_tb/num6 0 0 0 0 Image: /top_tb/num6 1 0 0 0 Image: /top_tb/num8 1 0 0 0 Image: /top_tb/num9 0 0 0 0 Image: /top_tb/num10 0 0 0 0 Image: /top_tb/num4 0 1 0 0 Image: /top_tb/num4 0 0 0 0 Image: /top_tb/num4 0 0 0 0 Image: /top_tb/num4 0 0 0 0 0 Image: /top_tb/num4 0 0 0 0 0 0 Image: /top_tb/cegnewhand 0 0 0 0 0 0 0 Image: /top_tb/cgbist 0 0 0 <td></td> <td></td> <td></td>			
Image: /top_tb/num4 0 Image: /top_tb/num5 0 Image: /top_tb/num6 0 Image: /top_tb/num7 1 Image: /top_tb/num9 0 Image: /top_tb/num10 0 Image: /top_tb/num4 0 Image: /top_tb/stenable 0 Image: /top_tb/stenable 0 Image: /top_tb/cgbist 0 Image: /t			
H / top_tb/num5 0 0 0 H / top_tb/num6 0 0 0 H / top_tb/num7 1 0 0 H / top_tb/num8 1 1 0 H / top_tb/num9 0 0 0 H / top_tb/num9 0 0 0 H / top_tb/num10 0 0 0 H / top_tb/num10 2 0 0 H / top_tb/num4 0 0 0 H / top_tb/reset 0 0 0 I / top_tb/global reset 0 0 0 0 I / top_tb/cgbist 0 0 0 0 0 I / top_tb/cgbist 0 0 0 0 0 0 I / top_tb/cgbist 0 0 0 0 0 0 0 I / top_tb			
Image: Comparison of the second se			
Image: Construction		• •	
Image: Arrow of the set of the se			
<pre> /top_tb/num10 0 2 2 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0</pre>			
Image: /top_tb/numj 2 Image: /top_tb/numq Image: /top_tb/numk			
<pre> /top_tb/numq /top_tb/numk /top_tb/enable /top_tb/reset /top_tb/bistenable /top_tb/globalreset /top_tb/globalreset /top_tb/cgnewhand /top_tb/cgbist Now ns 1700 Now Ns 17</pre>			
<pre>/top_tb/enable /top_tb/reset /top_tb/bistenable /top_tb/globalreset /top_tb/cgnewhand /top_tb/cgbist</pre>	⊡- <mark>∭</mark> /top_tb/numq	0 1 0 0	
<pre>/top_tb/reset 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</pre>			
<pre>/top_tb/bistenable /top_tb/globalreset /top_tb/cgnewhand /top_tb/cgbist</pre>			
<pre>/top_tb/global reset 0 /top_tb/cgnewhand /top_tb/cgbist Now ns Cursor 1 ns </pre>			
<pre>/top_tb/cgnewhand 0 // / / / / / / / / / / / / / / / / /</pre>			
Now ns 1700 1800 Cursor 1 ns 1700 1800	🗾 /top_tb/cgnewhand	o <u> </u>	
Now ns 1700 11100 1800 Cursor 1 ns 1200 12	/top_tb/cgbist		
Now ris 1700 1800 Cursor 1 ns 1700 1800			
		ns 1700 1800	
	Cursor 1		
	1639 ns to 1830 ns		

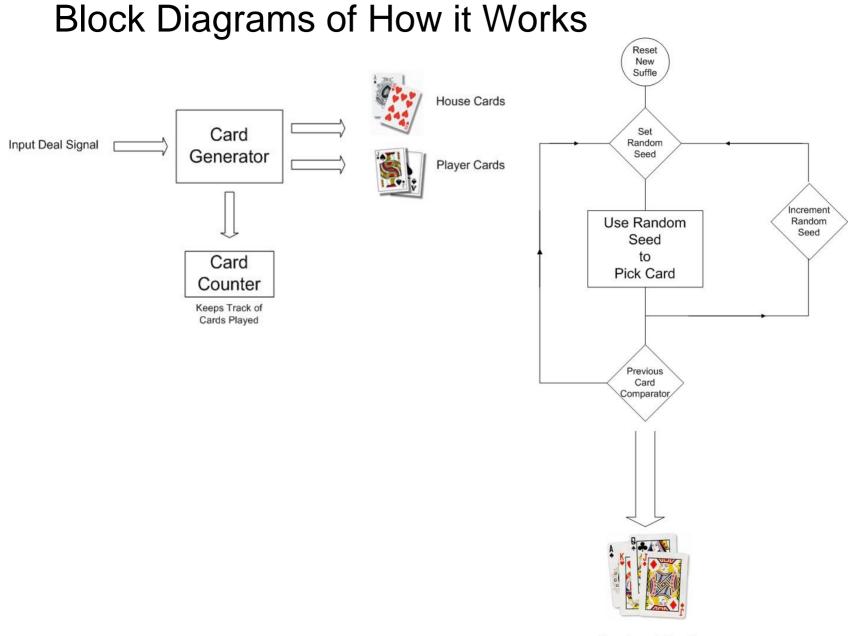
Probable errors:

- 'Action' vector needs to be only (n-1)*2 bits for 'n' players.
- •Ace being counted as 10 by the house player.

THANK YOU !!!!!

ECE 552 Spring 2005 Blackjack Project: Card Generator Group Ronnie Chai Eric Cardwell Nathan Dennis Pradeep Chimakurthy





Cards get Dealt

Card Generator Test Bench

File Edit View Insert Format Tools Window B 🕄 🚜 A → X 🔪 🖳 । 🔍 🔍 🔍 👫 📯 🖻 🔛 /card generator tb/uut/in clock 🧧 /card generator tb/uut/in reset 🧖 /card generator tb/uut/in issue 🔽 /card generator tb/uut/in bist /card generator tb/uut/in num twos 00000 /card_generator_tb/uut/in_num_threes 00000 **[+]**--/card_generator_tb/uut/in_num_fours 00000 /card_generator_tb/uut/in_num_fives 00000 /card generator tb/uut/in num sixes 00000 /card generator tb/uut/in num sevens 00000 /card_generator_tb/uut/in_num_eights 00000 [+]---/card_generator_tb/uut/in_num_nines 00000 **[]** ⊡-□ /card_generator_tb/uut/in_num_tens 00000 /card_generator_tb/uut/in_num_jacks 00000 00000 /card generator tb/uut/in num kings 00000 H-/card_generator_tb/uut/in_num_gueens /card_generator_tb/uut/in_num_aces 00000 0000 10001 10100 11101 20011 1001 11100 **H**-/card_generator_tb/uut/out_card /card_generator_tb/uut/out_house 1001 /1100 **⊞**–, 🔽 /card generator tb/uut/out issued 1 1 1 00 ns Now. 500 1 us 1500 Cursor 1 0 ns l0 ns 0 ns to 1700 ns Now: 2 us Delta: 0

Card Counter Test Bench

File Edit View Insert Format Tools Window B 🖪 🖊 ▶ 👗 1+ → N 🖪 I 🔍 🔍 🔍 👫 34 H 2 /cardcounter tb/uut/clock /cardcounter tb/uut/reset /cardcounter tb/uut/issued /cardcounter_tb/uut/card 1101 10010 1011 10001 <u>looon</u> /cardcounter tb/uut/stat ace 00000 00000 loooot /cardcounter_tb/uut/stat_two **H**-/cardcounter_tb/uut/stat_three 00000 /cardcounter_tb/uut/stat_four 00000 /cardcounter tb/uut/stat five 00000 /cardcounter tb/uut/stat six 00000 /cardcounter_tb/uut/stat_seven 00000 [+]---/cardcounter_tb/uut/stat_eight 00000 ⊡-□ /cardcounter tb/uut/stat nine 00000 /cardcounter tb/uut/stat ten 00000 /cardcounter_tb/uut/stat_jack 00000 <u>looooh</u> 100010 **H**-/cardcounter_tb/uut/stat_queen 00000 /cardcounter tb/uut/stat king 00000 <u>looooh</u> 00000 <u>Xooooh</u> **H**-/cardcounter tb/uut/ace <u>Yoooot</u> /cardcounter tb/uut/two 00000 /cardcounter_tb/uut/three 00000 /cardcounter_tb/uut/four 00000 [**+**]--/cardcounter_tb/uut/five 00000 00000 /cardcounter_tb/uut/six /cardcounter_tb/uut/seven 00000 /cardcounter_tb/uut/eight 00000 /cardcounter_tb/uut/nine 00000 [**+**]--| /cardcounter_tb/uut/ten 00000 **H**-/cardcounter_tb/uut/jack 00000 <u>looooh</u> ľ0001b [+]--/cardcounter tb/uut/queen 00000 ⊡-□ /cardcounter_tb/uut/king 00000 <u>looooh</u> 00 ns Now. Cursor 1 0 ns l0 ns 0 ns to 230 ns Now: 2 us Delta: 0

Problems Identified

- First card for dealer does not increment card counter as out_issued flag not set.
- Card Sequence correction can be problematic based on random seed and sequence if(card > 12) then

```
card := (14 - card) + sequence;
When a card generated has upper 4 bits 1111 and sequence is 0,
taking the card := card + 1; generates a card that has value 0.
```

Logic to ensure that card generated does not exceed pack is faulty.

```
if( sum_of_played = 16 ) then
    if( tmp_card = "1100 (Queen)" ) then
tmp_card := (others=>'0');--did not consider card value '0' in switch case
    else
    tmp_card := tmp_card + 1;
    end if;
```

Initialization of signals which is problematic with synthesis tools.

Code Improvements

<u>Why?</u>

- Synthesis problems
- Randomness of card generation
- Accounting cards between games

How?

- Introduce reset
- Use memory cell stacks.
- Shuffle deck after 2 decks or 104 cards are dealt.

Test Bench Improvements

Why?

- •Coverage for card_generator.vhd is only 75%
- •Coverage for cardcounter.vhd is only 68.8%
- •Coverage should be at least 95%

How?

- •Increase the duration of signals in both test benches
- •Look for other combinations of input



ECE 552 Black Jack Presentation

Score Keeper & Banker

552 Black Jack (Score Keeper & Banker) - 1



Score Keeper:

Robert Greenwell Derek Ziemian

552 Black Jack (Score Keeper &Banker) – 2

Score Keeper – What Does It Do?

- Receives information from the Controller pertaining to
 - Which player is currently on turn
 - When and what card that player has received
 - When all players are standing (hand)
- This information is used to compile a score based on
 - How many cards a player has
 - And the value of said cards
 - A value of 21 with 2 cards = Blackjack
 - A value of 21 = Win (unless BJ or tie)
 - If there is no Blackjack or 21, the hands are compared to see who wins and if there is a tie
- Next, information is sent to the Banker
 - A signal telling the Banker the score is final
 - And a matrix containing each players stats (Blackjack, Win, Tie, or Lose)



Score Keeper – The Code

- While the code for the Score Keeper was for most the part functional, there were several areas for improvement
 - The code was poorly commented. Comments have been added to the code as necessary to understand its flow
 - There was a large amount of unused code in the program (such as unutilized loops and variables), and this code has been removed as deemed fit
 - An internal vector (HP_BJ) was not being reset, and if tripped, would cause the score to be incorrectly reported to the Banker in any subsequent hands. This error has been corrected



Score Keeper – Test Bench

- There where issues found with the preexisting Test Bench for the Score Keeper
 - Only a very simplistic simulation was conducted by the existing Test Bench
 - A new Test Bench was created to provide for more rigorous simulations
 - Added additional vectors to test the functionality of a larger portion of the code
 - Added the ability to simulate 6 Players instead of only 3



Banker:

Steven Bunch Lanie Britton Chris Beall

552 Black Jack (Score Keeper & Banker) – 6

Banker Code Tasks - What Does It Do?

- Initializes each player's bank account to 127
 - Then manipulates an individual's account
 - Tracks bets wagered
 - Subtracts or adds money in an account due to wins or losses
- Monitors the score keeper for the end of the game to be reported
 - Monitors for either 100 hands played, or
 - If a player has reached the max bank allowed, or
 - If a player is out of money
- Lets the controller know that the game has ended
 - A signal out indicates that the controller should end the game or playing session



- As with the majority of the Black Jack code, the Banker code was poorly commented
- Although generally functional, many signals in the code were unused
- Other sections of the code might also be determined to be unused, but will require modification of the testbench



Modifications/Suggestions

- The code was commented for ease of reuse
- Bulky unused code was removed
- The testbench will need to be modified
 - Ensure Functionality
 - Ensure Code Coverage