

Parking System

Proyek Akhir PSD



Kelompok BP03

Anggota Kelompok

- 01 2206059401 Nakita Rahma Dinanti
- 02 2206809841 Kamal Makarim Iskandar
- 03 2206813353 Naufal Rusyda Santosa
- 04 2206814186 Fathin Umara Aero

Deskripsi Parking System

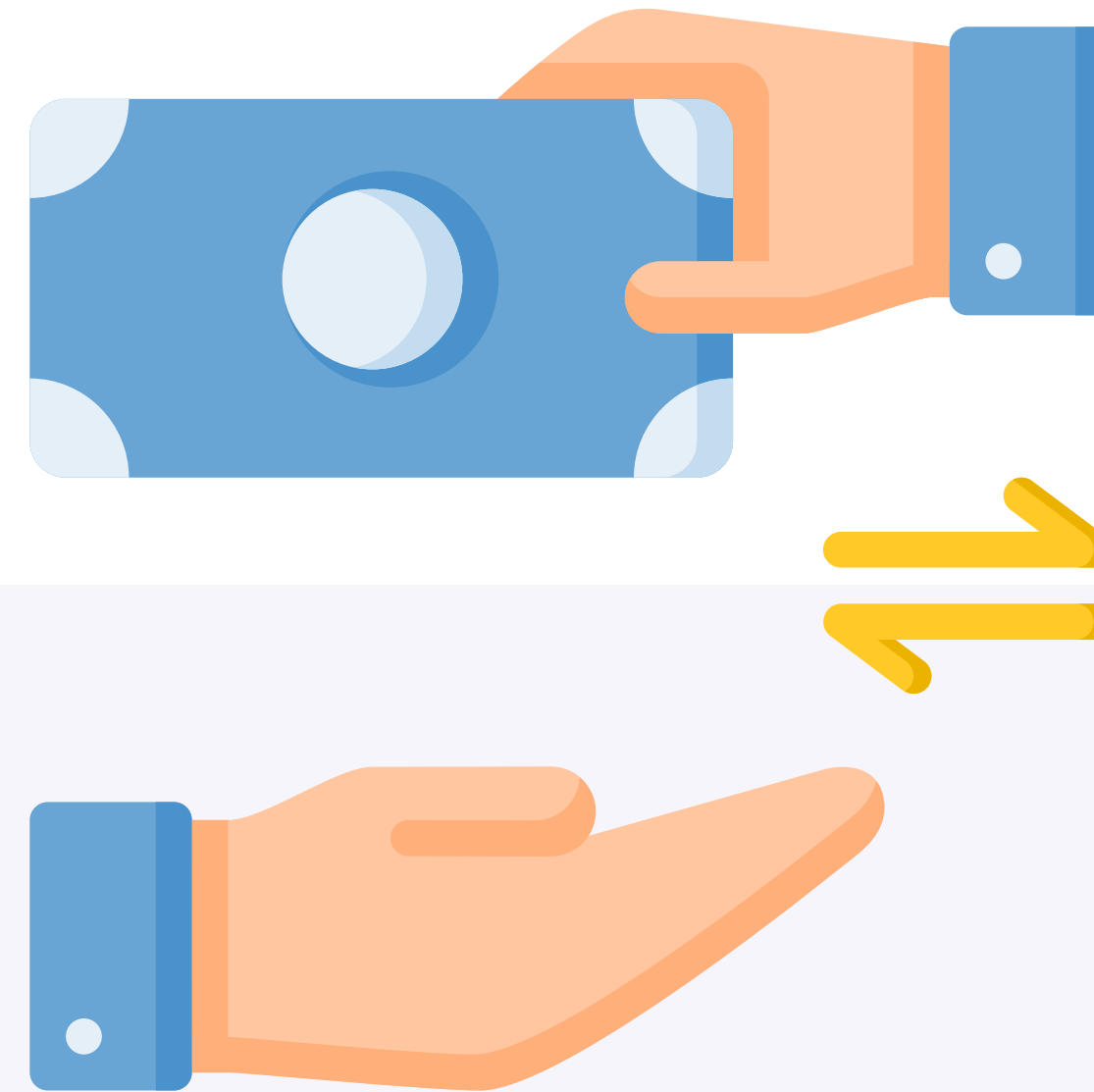
Proyek kami disini akan merepresentasikan parking system dimana ada 3 unit yang akan digunakan, yaitu **Memory**, **Interface Masuk**, **Interface Keluar** dan **CPU**.

Memory akan menyimpan kapan waktu masuk, ID kartu, dan balance dari kartu. **Inteface Masuk** akan menerima id kartu, menghitung kapan waktu masuk dari clock, kemudian menyimpannya di memory. **Interface Keluar** akan menerima id kartu dan balance, mengeluarkan balance baru ketika balancenya cukup, kalau balance kurang palang akan tetap tertutup. **CPU** akan mengatur semua alur kegiatan.



Tujuan Proyek

Tujuan kami dengan membuat **Parking System** ini adalah untuk memudahkan proses pembayaran parkir dengan menggunakan kartu e-Money. Serta kita juga dapat mengetahui berapa lama waktu parkir.





Unit Kerja

CPU

CPU adalah unit yang mengatur dan menjaga segala kegiatan agar tetap berjalan lancar. CPU memiliki total 7 buah instruksi yang masing-masing terdiri dari 3-bit opcode.

Registers masing-masing 64 bit		
Reg Number	bit	Reserved for
0	000	Timestamp
1	001	Used parked spot
2	010	-
3	011	-
4	100	-
5	101	-
6	110	-
7	111	-

Instructions				
No	Name	Opcode	Bits	data_in
1	remove	000	Opcode (3bit)	(31 downto 0) id kartu
2	sub RA, RB	001	Opcode (3bit) + mode(2 bit) + Oppoperand A(3 bit) + Oppoperand B (3 bit)	Tergantung mode
3	mul RA, RB	010	Opcode (3bit) + mode (2 bit) + Oppoperand A(3 bit) + Oppoperand B (3 bit)	Tergantung mode
4	mov A, B	011	Opcode (3bit) + mode(2 bit) + Oppoperand A(3 bit) + Oppoperand B (3 bit)	Tergantung mode
5	save	100	Opcode (3bit)	
6	div RA, RB	101	Opcode (3bit) + mode (2 bit) + Oppoperand A(3 bit) + Oppoperand B (3 bit)	(31 downto 0) new time
7	Inc RA	110	Opcode (3bit) + dont care (2 bit) + Oppoperand A (3 bit)	
8	Dec RA	111	Opcode(3bit) + dont care (2bit) + Oppoperand A (3 bit)	

Mode (mov)				
no	mode	bit	A (3 bit)	B (3 bit)
1	data_out, reg	00	XXX	Reg B
2	reg, mem	01	Reg A	Mem[Reg B]
3	reg, data_in	10	Reg A	XXX
4	mem, reg	11	Mem[Reg A]	Reg B

Mode (aritmatika)				
no	mode	bit	A (3 bit)	B (3 bit)
1	data_out, reg	00	XXX	Reg B
2	reg, mem	01	Reg A	Mem[Reg B]
3	reg, data_in	10	Reg A	XXX
4	reg, reg	11	Reg A	Reg B

Interface Masuk

Pada unit **Interface Masuk**, kami membuat program yang akan membaca ID dari kartu pengguna yang hendak parkir. Kemudian program akan membaca waktu masuk menggunakan Unix Timestamps dengan format Year-Month-Day-Hour-Minute-Second.

Selanjutnya, Card ID dan Timestamps akan digabungkan dan disimpan ke dalam **Memory**

	M <= Card id(32 bit) + timetstamp (32bit)	Instruction :
Get_timestamp	MOV data_out, R0	011 00 000 000
cardid_timestamp	datain(63 downto 32) <= data_out	
	datain(31 downto 0) <= card_id	
move_toreg	MOV R2, data_in	011 10 010 000
move_tomem	MOV M[R1], R2	011 11 001 010
	Inc R0	110 001 00000

Memory

Unit **Memory** berfungsi untuk menyimpan Card ID dan waktu scan pada **Interface Masuk** serta menyimpan balance kartu ketika di-scan pada **Interface Keluar**.

Interface Keluar

Unit **Interface Masuk** akan membaca kembali CardID dari pengguna yang hendak keluar, kemudian membaca balance dari kartunya. Apabila balance nya cukup, maka akan dipotong sesuai dengan tarif parkir.

	interfaceKeluar	
set_0_r2	data_in <= others 0	
	mov R2, data_in	011 10 010 000
card id to data_in	data_in(31 downto 0) <= card_id	
mov R3 data_in	mov R3, data_in	011 10 011 000
mov R4 M[R2]	mov R4, M[R2]	011 01 100 010
mov data_out R4 1	mov data_out, R4	011 00 000 100
remove timestamp	data_in(31 downto 0) <= data_out(63 downto 32)	
	data_in(63 downto 32) <= others 0	
mov R4 data_in	mov R4 data_in	011 10 100 000
sub R4 R3	sub R4, R3	001 11 100 011
mov data_out R4 2	mov data_out, R4	011 00 000 100
check data_out	if data_out = 0, next_state = mov R3 MR2	
	else, inc R2 ,next_state = mov R4 M[R2]	110 010 00000
mov R3 MR2	mov R3, M[R2]	011 01 011 010
mov data_out R3	mov data_out R3	011 00 000 011
remove cardid	data_in(63 downto 32) <= others 0	
	data_in(31 downto 0) <= data_out(31 downto 0)	
mov R3 data_in	mov R3, data_in	011 10 011 000
get_timestamp	mov data_out , R0	011 00 000 000
mov R4 data_in	data_in <= data_out	
	mov r4, data_in	011 10 100 000
sub R4 R3	sub R4, R3	001 11 100 011
div R4, data_in	data_in <= 60 * 60	
	div R4, data_in	101 10 100 000
mul R4 data_in	data_in <= 3000	
	mul R4, data_in	010 10 100 000
mov balance to reg	data_in <= balance	
	mov R5, data_in	011 10 101 000
sub R5 R4	sub R5, R4	001 11 101 100
mov data_out R5	mov data_out R5	011 00 000 101
check_balance	if data_out >= 0, balance<=data_out	
	else report uang tidak cukup	
save		100 00000000



Simulasi & Sintesis RTL

Sintesis RTL

Simulasi ModelSim

+	/computer/dlk	1	
+	/computer/data_in	UUUUUUUUUUUU...	
+	/computer/data_out	UUUUUUUUUUUU...	
+	/computer/opcode	UUU	
+	/computer/mode	UU	
+	/computer/operandA	UUU	
+	/computer/operandB	UUU	
+	/computer/reg	{0000000000000000...	
+	/computer/addr	UUUUUUUU	
+	/computer/data_in_memory	UUUUUUUUUUUU...	
+	/computer/data_out_memory	UUUUUUUUUUUU...	
+	/computer/balance	UUUUUUUUUUUU...	
	/computer/done	U	
	/computer/wir_en	U	
	/computer/state	s0	
	/computer/shift	U	
+	/computer/instruction1	UUUUUUUUUU	
+	/computer/instruction2	UUUUUUUUUU	
+	/computer/data_in1	UUUUUUUUUUUU...	
+	/computer/data_in2	UUUUUUUUUUUU...	
+	/computer/data_out1	UUUUUUUUUUUU...	
+	/computer/data_out2	UUUUUUUUUUUU...	
	/computer/done1	U	
	/computer/done2	U	
	/computer/want_to_use1	U	
	/computer/want_to_use2	U	
	/computer/cpu_not_ready1	U	
	/computer/cpu_not_ready2	U	
	/computer/year1	-2147483648	
	/computer/year2	-2147483648	
	/computer/month1	-2147483648	
	/computer/month2	-2147483648	
	/computer/day1	-2147483648	
	/computer/day2	-2147483648	
	/computer/hour1	-2147483648	
	/computer/hour2	-2147483648	
	/computer/minute1	-2147483648	
	/computer/minute2	-2147483648	
	/computer/second1	-2147483648	
	/computer/second2	-2147483648	
+	/computer/card_id1	UUUUUUUUUUUU...	
+	/computer/card_id2	UUUUUUUUUUUU...	
+	/interfacekeluar/INSTRUCTION	UUUUUUUUUU	
+	/interfacekeluar/DATA_IN	UUUUUUUUUUUU...	
	/interfacekeluar/DONE	U	
	/interfacekeluar/WANT_TO_USE	U	
	/interfacekeluar/CLK	U	
	Now	0 ps	ps
	Cursor 1	0 ps	0 ps

Terima Kasih

