

Project:	Employ the <i>Unix</i> utility lex to create a lexical analyzer for the P-Code Assembler. Reserve words, mnemonics, numeric constants, and other token descriptions are described later in this document. Produce a trace file having the suffix “.atrc” and containing a description of every token encountered in the source file. The contents of a sample trace file are shown in Figure 3.	
Program Files:	File	Description
	makepasm	File makepasm contains instructions for program pasm . Instructions are written for the <i>Unix</i> utility make . Program pasm is contained in file pasm .
	mkpasm	File mkpasm is a Unix script file that removes old file created in the last creation of executable file pas and invokes file makepasm to create a new executable file pasm .
Command Line:	Project p04.1 can be invoked with zero or one program parameters. The first program parameter is the input file name. Sample command lines together with corresponding actions by program pasm are shown below. Boldfaced type indicates data entered at the keyboard by the user. \$ pasm Enter the input file name: p00.pasm \$ pasm p00.pcd	
Input File:	The input file contains a P-Code Assembler program. The input file name must have the suffix .pcd. Please refer to figure 1 for an example of the format of an input file.	
Output Files:	A single output file is produced. The output file has the same prefix as the input source.pcd and the suffix .pcd is replaced by the suffix .atrc. For example, if the input file was named hello.pcd, the output file would be named hello.atrc. \$ pasm p00.pcd File p00.atrc is produced as shown in Figure 2.	

L00001	ent	sp	L00002
	ent	ep	L00003
	rtn	p	
#define	L00002	4	
#define	L00003	4	
	mst	0	
	cup	0	L00001
	stp		

Figure 1. Example input file **p00.pcd**.

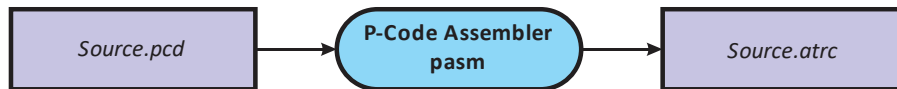


Figure 2. P-Code Assembler Block Diagram

```
Token=(92, LABEL, L00001)
Token=(4, MST_0, ent)
Token=(79, EP_R, sp)
Token=(92, LABEL, L00002)
Token=(4, MST_0, ent)
Token=(80, MP_R, ep)
Token=(92, LABEL, L00003)
Token=(6, EQU_0, rtn)
Token=(91, X_T, p)
Token=(93, DEFINE, #define)
Token=(92, LABEL, L00002)
Token=(94, INTLIT, 4)
Token=(93, DEFINE, #define)
Token=(92, LABEL, L00003)
Token=(94, INTLIT, 4)
Token=(5, RTN_0, mst)
Token=(94, INTLIT, 0)
Token=(2, CSP_0, cup)
Token=(94, INTLIT, 0)
Token=(92, LABEL, L00001)
Token=(50, LDA_0, stp)
```

Figure 3. Example Output file *p00.atrc*.

Token Code	Token Name	Pattern	Token Code	Token Name	Pattern
1	CUP_O	cup	46	RND_O	rnd
2	CSP_O	csp	47	CHR_O	chr
3	ENT_O	ent	48	ORD_O	ord
4	MST_O	mst	49	STP_O	stp
5	RTN_O	rtn	50	LDA_O	lda
6	EQU_O	equ	51	LDC_O	ldc
7	NEQ_O	neq	52	LDI_O	ldi
8	GRT_O	grt	53	LVA_O	lva
9	GEQ_O	geq	54	LVB_O	lvb
10	LES_O	les	55	LVC_O	lvc
11	LEQ_O	leq	56	LVI_O	lvi
12	ADI_O	adi	57	LVR_O	lvr
13	SBI_O	sbi	58	LVS_O	lvs
14	NGI_O	ngi	59	LVT_O	lvt
15	MPI_O	mpi	60	STI_O	sti
16	DVI_O	dvi	61	IXA_O	ixa
17	MOD_O	mod	62	RDB_F	rdb
18	ABI_O	abi	63	RDC_F	rdc
19	SQI_O	sqi	64	RDI_F	rdi
20	INC_O	inc	65	RDR_F	rdr
21	DEC_O	dec	66	RLN_F	rln
22	ADR_O	adr	67	WRB_F	wrb
23	SBR_O	sbr	68	WRC_F	wrc
24	NGR_O	ngr	69	WRI_F	wri
25	MPR_O	mpr	70	WRE_F	wre
26	DVR_O	dvr	71	WRF_F	wrf
27	ABR_O	abr	72	WRS_F	wrs
28	SQR_O	sqr	73	WRT_F	wrt
29	IOR_O	ior	74	WLN_F	wln
30	AND_O	and	75	SQT_F	sqt
31	XOR_O	xor	76	LN_F	ln
32	NOT_O	not	77	EXP_F	exp
33	INN_O	inn	78	SP_R	sp
34	UNI_O	uni	79	EP_R	ep
35	NTR_O	ntr	80	MP_R	mp
36	DIF_O	dif	81	PC_R	pc
37	CMP_O	cmp	82	NP_R	np
38	SGS_O	sgs	83	A_T	a
39	UJP_O	ujp	84	B_T	b
40	XJP_O	xjp	85	C_T	c
41	FJP_O	fjp	86	I_T	i
42	TJP_O	tjp	87	R_T	r
43	FLT_O	flt	88	S_T	s
44	FLO_O	flo	89	T_T	t
45	TRC_O	trc	90	P_T	p

Token Code	Token Name	Pattern	Token Code	Token Name	Pattern
91	X_T	x			

Token Code	Token Name	Pattern
92	LABEL	A label begins with the capital letter L and is followed by any number of integer digits. Example: L00001
93	DEFINE	#define
94	INTLIT	An integer literal is an optional sign followed by one or more digits.
95	REALIT	A real literal – a real number constant – consists of one or more integer digits, a decimal point, one or more fractional digits, and an optional exponent. The exponent, if present, is the letter e, an optional sign, and one or more digits. Remember, P-Code is case-insensitive so the letter e may be capitalized.
95	REALIT	A real literal – a real number constant – consists of one or more integer digits and an exponent.
96	CHRLIT	A character constant – a single character enclosed between apostrophes. For example, the character constant, t, appears in the source as 't'. An apostrophe is double so that a single apostrophe appears as ''.
97	STRLIT	A string constant – two or more characters enclosed between apostrophes. For example, the string constant banana appears as 'banana'. Apostrophes in string constants are doubled so that the constant don't appears as 'don''t'.
98	ID	Identifier – One or more lower case letters
99	ERROR	An unrecognized token