

Label	Location	Contents	Mnemonic	Operand	Comment
	ROM:				
	027	907C	LDM	BRetInst	Base Return Inst (JPI RetAdd1)
	028	A804	STM	RetInst	
	029	804D	LDI	04D	UART mode with undivided clk
	02A	AC04	STM	UART Control	
	02B	8037	LDI	037	UART command instruction
	02C	AC04	STM	UART Control	
	02D	8FFF	LDI	FFF	Initial instruction index (-1)
	02E	A807	STM	InstIndex	
MainLoop	02F	8004	LDI	004	Initial Nybble no. (loop 4 times)
	030	A809	STM	NybbleNo	
ReturnLoop	031	9809	LDM	NybbleNo	
	032	2079	SUB	One	
	033	E03F	JPM	Done	Instruction complete; store in RAM
	034	A809	STM	NybbleNo	Instruction not complete;
	035	007D	ADD	BLDMMask	Get nybble mask by creating indexed
	036	A803	STM	IndInst	instruction
	037	803A	LDI	03A	
	038	A805	STM	RetAdd1	
	039	B803	JMP	IndInst	Jump to indexed instruction
	03A	A808	STM	Mask	Store mask retrieved from table
	03B	803E	LDI	03E	
	03C	A806	STM	RetAdd2	Store return address
	03D	B04E	JMP	RxPoll_1	Get next char and process
	03E	B031	JMP	ReturnLoop	
Done	03F	9807	LDM	InstIndex	Increment index and store
	040	0079	ADD	One	
	041	A807	STM	InstIndex	
	042	007E	ADD	BSTMInst	Create indexed store instruction
	043	A803	STM	IndInst	
	044	8048	LDI	048	Return address
	045	A805	STM	RetAdd1	
	046	9802	LDM	Inst	
	047	B803	JMP	IndInst	Store instruction in RAM
TxLoop_2	048	9C04	LDM	UART Control	
	049	4079	AND	One	Check TxRdy
	04A	D048	JPZ	TxLoop_2	
	04B	800D	LDI	ASCII CR	Carriage return
	04C	AC03	STM	UART Data	Send CR to display
	04D	B02F	JMP	MainLoop	
RxPoll_1	04E	9C04	LDM	UART Control	
	04F	407A	AND	Two	Check RxRdy

	050	D04E	JPZ	RxPoll_1	
	051	9C03	LDM	UART Data	Get Char
	052	A800	STM	Char	Store char
	053	206B	SUB	CTRL-C	End of input?
	054	D80A	JPZ	ProgStart	Yes-start program
TxPoll_1	055	9C04	LDM	UART Control	No- Echo Char and process
	056	4079	AND	One	Check TxRdy
	057	D055	JPZ	TxPoll_1	
	058	9800	LDM	Char	
	059	AC03	STM	UART Data	Echo Character
	05A	207B	SUB	30hex	Make index by subtracting 30h
	05B	007F	ADD	BLDMInst	Create indexed LDM instruction
	05C	A803	STM	IndInst	Store in RAM
	05D	8060	LDI	060	Return address
	05E	A805	STM	RetAdd1	
	05F	B803	JMP	IndInst	Get partial inst from table
	060	A801	STM	PartInst	
	061	9808	LDM	Mask	
	062	D067	JPZ	Next1	Special for highest nybble
	063	4801	AND	PartInst	Mask off partial inst
	064	0802	ADD	Inst	Combine with prior inst and store
	065	A802	STM	Inst	
	066	C806	JPI	RetAdd2	Done
Next1	067	4801	AND	PartInst	Mask off partial inst
	068	A802	STM	Inst	Store (high nybble)
	069	C806	JPI	RetAdd2	Done
Zero	06A	0000			
CTRL-C	06B	0003			
MaskTable	06C	000F			
	06D	00F0			
	06E	0F00			
InstTable	06F	0000			
	070	1111			
	071	2222			
	072	3333			
	073	4444			
	074	5555			
	075	6666			
	076	7777			
	077	8888			
	078	9999			
One	079	0001			
Two	07A	0002			

30hex	07B	0030			
BRetInst	07C	C805	JPI	RetAdd1	
BLDMMask	07D	906C	LDM	MaskTable	
BSTMInst	07E	A80A	STM	ProgStart	
BLDMInst	07F	906F	LDM	InstTable	
	080	AAAA			
	081	BBBB			
	082	CCCC			
	083	DDDD			
	084	EEEE			
	085	FFFF			
	08E	907C	LDM	BRetInst	
	08F	A804	STM	RetInst	
	090	804E	LDI	04E	UART mode with Rx and Tx clk/16
	091	AC04	STM	UART Control	
	092	B02B	JMP	02B	Entry with UART clk/16
Label	Location	Contents	Mnemonic	Operand	Comment
	RAM:				
Char	800				
PartInst	801				
Inst	802				
IndInst	803				
RetInst	804				
RetAdd1	805				
RetAdd2	806				
InstIndex	807				
Mask	808				
NybbleNo	809				
ProgStart	80A				