Due on: November 24, 2022

Introduction to Microcomputers

Lab4

The goal of this lab is to make use of function calls and indirect addressing using PIC16F877A assembly instructions.

Assignment

Consider an arbitrary number generator function GenerateNumbers that takes 3 arguments, x, y and N, and generates some arbitrary numbers in a loop, and stores these numbers in an array. It then returns the total number of elements generated and stored in the array. You then write another function AddNumbers that adds the numbers in the array and returns their sum. Finally, you write a function DisplayNumbers that first displays the sum and then the first 5 numbers in the array one after the other as the user presses Button3 connected to PORTB3 (RB3 on PICSIM).

Here is the sketch of the C code that you need to implement in PIC16F877A:

```
uint8_t Multiply(uint8_t x, uint8_t y) {
	uint16_t z = x * y;
	uint8_t* p = (uint8_t*)&z;
	return 2 * p[0] + p[1];
} //end-Multiply

uint8_t GenerateNumbers(uint8_t A[], uint8_t x, uint8_t y, uint8_t N) {
	int8_t count = 0;
	while ((x < N) || (y < N)) {
		if ((x + y) % 2 == 1) {
			A[count++] = Multiply(x, y);
			x = x + 1;

		}
		else {
			uint8_t tmp = x + y;
			A[count++] = tmp / 3;
			y = y + 3;
		} // end-else
	} //end-while

	return count;
} //end-GenerateNumbers

uint8_t AddNumbers(uint8_t A[], uint8_t count) {
	uint8_t sum = 0;
	for (int i = 0; i < count; i++) {
		sum += A[i];
	} //end-for

	return sum;
} //end-AddNumbers

void DisplayNumbers(uint8_t sum, uint8_t A[]) {
	BANKSEL TRISD;		// Select Bank1
	TRISD = 0;		// All pins output
	TRISB = 0xFF;		// All pins input
```

```
	BANKSEL PORTD;		// Select Bank0
	PORTD = sum;
	while (PORTB3 == 1);	// Wait until the user presses RB3;

	for (int i = 0; i < 5; i++) {
		PORTD = A[i];
		DelayMs(250);		// Wait for 250 ms
		while (PORTB3 == 1);	// Wait until the user presses RB3;
	} //end-for
} //end-DisplayNumbers

void main() {
	uint8_t x = 112;
	uint8_t y = 100;
	uint8_t N = 125;
	uint8_t A[40];
	uint8_t noElements = GenerateNumbers(A, x, y, N);
	uint8_t sum = AddNumbers(A, noElements);
	DisplayNumbers(sum, A);
	while (1);
}//end-main
```

The expected result for some x, y and N values are given in the following table:

x	y	N	sum	A[] (First 5 numbers)
112	100	125	129	[70, 77, 72, 194, 73]
200	177	213	56	[26, 40, 53, 42, 92]
224	211	235	109	[248, 60, 232, 61, 227]
7	11	23	56	[6, 196, 7, 16, 8]