

## 第五章 小腦模型控制器應用於超音波馬達定位控制

本篇論文使用 Microchip 公司所生產的 PIC18F452 微控制器為 CMAC 的核心，除了不需使用 EEPROM，其它所需使用到的特殊功能大致與 RBC 相同，因在第四章已詳細介紹過此微控制器的特殊功能，此章節將不再重述。

### 5.1 小腦模型控制器應用於 LUSM 定位控制

如圖 5-1 所示，為 CMAC 應用於 LUSM 定位控制方塊圖，RBC 會不斷去送出控制電壓( $v_0$ )至 Driver，當  $e(n)=0$  時，LUSM 的期望值位置( $\bar{y}_d$ )等於實際值位置( $\bar{y}_s$ )，則控制電壓為 0。

除了控制方法改為小腦模型理論之外，方塊圖內容大致與 RBC 相同，因本控制器使用 2 維學習空間來學習 LUSM 的數學模型，其最大的優點可以依照期望值電壓的變化率大小來修正 LUSM 移動速度的大小，所以加法器必須能夠儲存上一次誤差信號的值，將有兩組誤差信號( $e(n)$ 、 $e(n-1)$ )輸入至 CMAC，其詳細過程將在 5.3 節中說明。

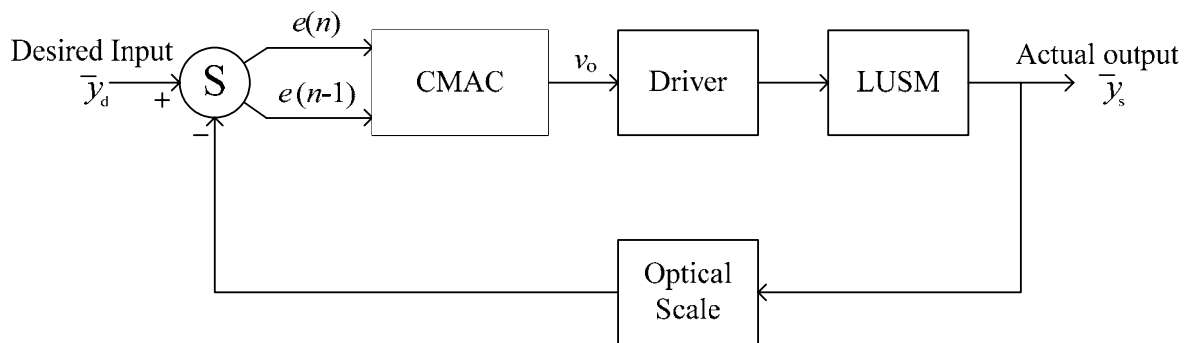


圖5-1 CMAC 應用於 LUSM 定位控制方塊圖

## 5.2 小腦模型控制器的製作流程

本控制器的製作流程最主要可分離線學習與線上控制兩大部分，離線學習需先知道 USM 的數學模型，而線上控制是為了驗證 USM 數學模型是否正確。

### 5.2.1 離線(Off-line)學習

離線學習的步驟如下所示：

1. 使用先前所製作 RBC 來控制超音波馬達的定位，大約繪製出 LUSM 的數學模型；首先將 LUSM 定位的距離作區段分割，如表 5-1 所示，再將此距離與輸出的控制電壓顯示在 LCD 上，最後紀錄下超音波馬達的  $l(n)$  與控制電壓並繪製成曲線圖(空載)，如圖 5-2 所示。
2. 決定期望值量化的等份(16 等份)、輸出至 LUSM 驅動器的控制電壓大小(0~5V)、與此控制電壓可調變的最高解析度(255 等份, 19.6mV)、CMAC 的學習空間、量化層的數量(2 層)與參考狀態的個數(256)。為了加強 CMAC 對超音波馬達定位的性能與適應能力，本篇論文將 CMAC 規劃成二維學習空間，可以使 LUSM 在數學模型或負載改變時，CMAC 依然有良好的控制性能。
3. 使用 MATLAB 軟體在個人電腦進行模擬，計算出每一個記憶體的權值，並將每一個記憶體權值，直接寫入至程式記憶體中，再使用組合語言透過 Microchip 公司所提供的 MPLAB 軟體編譯成機械碼燒錄至 PIC18F452。

表5-1 定位距離對應區段分割表

$l(n)$	0	1	2	3	4	5	6	7
Length(min)	0	1	256	384	512	640	768	896
Length(max)	0	255	383	511	639	767	895	1023
$l(n)$	8	9	10	11	12	13	14	15
Length(min)	1024	1152	1280	1408	1536	1664	1792	1920
Length(max)	1151	1279	1407	1535	1663	1791	1919	2047

$l(n)$ : 超音波馬達起點與終點相對的距離

$n$ : 現在的狀態

$v_0$ : RBC 輸出至 LUSM 驅動器的控制電壓

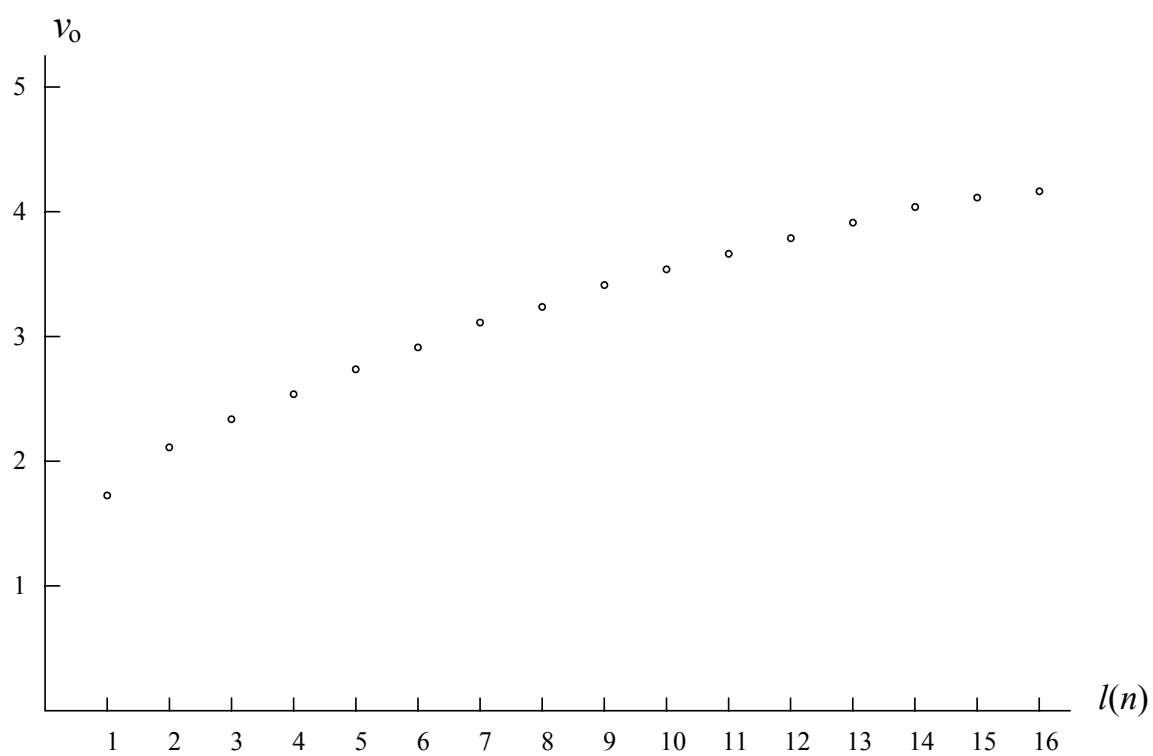


圖5-2 LUSM 的  $l(n)$  對應控制電壓曲線圖(空載)

## 5.2.2 線上控制

完成離線學習後，CMAC 即可對 LUSM 做線上控制，控制的方法可分 3 大部份，主程式、Timer 計數與中斷副程式，除了主程式部分與 RBC 不同外，其餘皆與 RBC 相同，在此節將不再重述，只針對主程式部分加以說明。

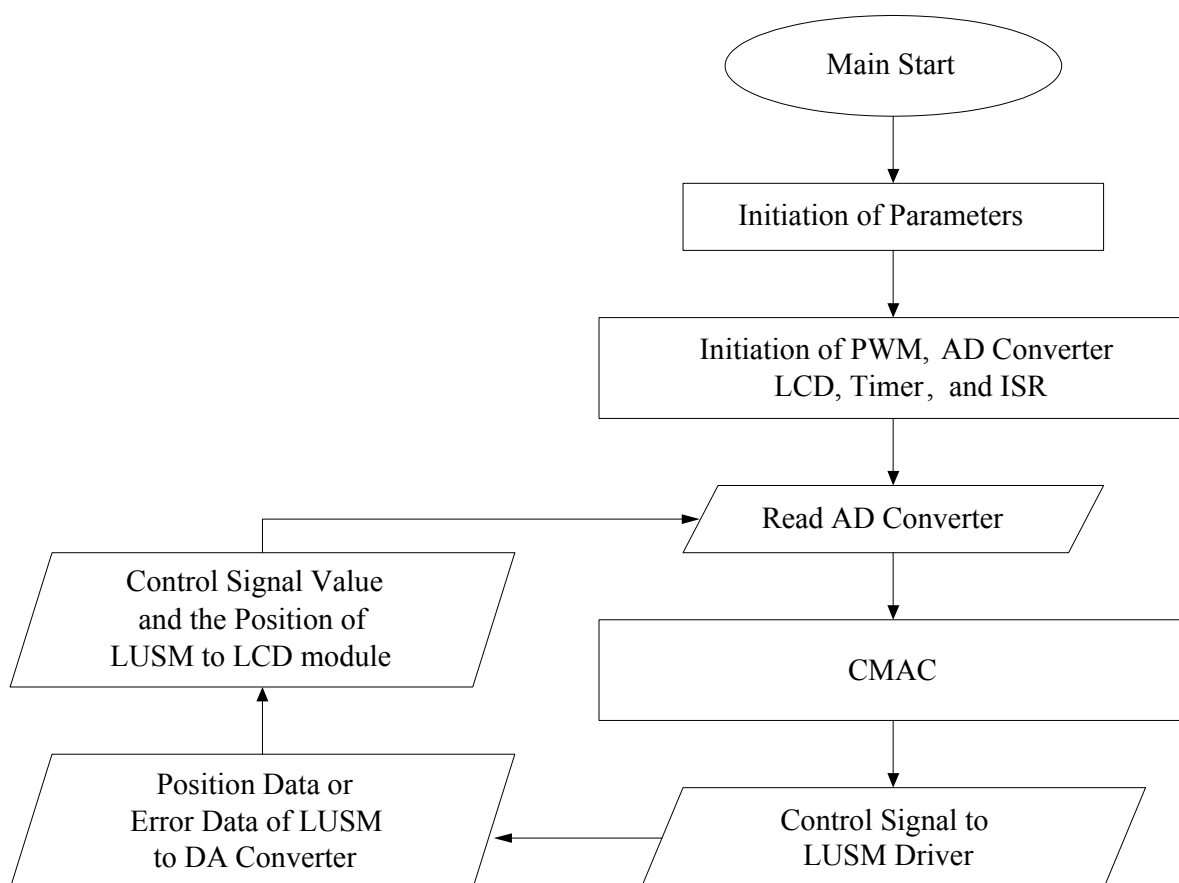


圖5-3 CMAC 處理流程圖

圖 5-3 為 CMAC 主程式流程圖，除了控制方法與 RBC 不同外，其餘皆與 RBC 相同，而 CMAC 控制 LUSM 流程如圖 5-4 所示，可分為 4 個步驟，摘要如下：

1. Read  $l(n)$  and  $l(n-1)$ : 讀取  $l(n)$  與  $l(n-1)$  的值，並對應至分割的區塊號碼。
2. Correspond to Memory with Index Vector: 將區塊號碼對應至記憶體索引向量，計算出實體記憶所在的位址。

3. Add Values of Memory：將所對應之記憶體的內容相加，輸出至 LUSM 驅動器。
4. Control Signal to LUSM Driver：如附錄二[26]所示，LUSM 驅動器的  $(VIN+) > (VIN-)$ ，LUSM 向右移動；LUSM 驅動器的  $(VIN+) < (VIN-)$ ，LUSM 向左移動。

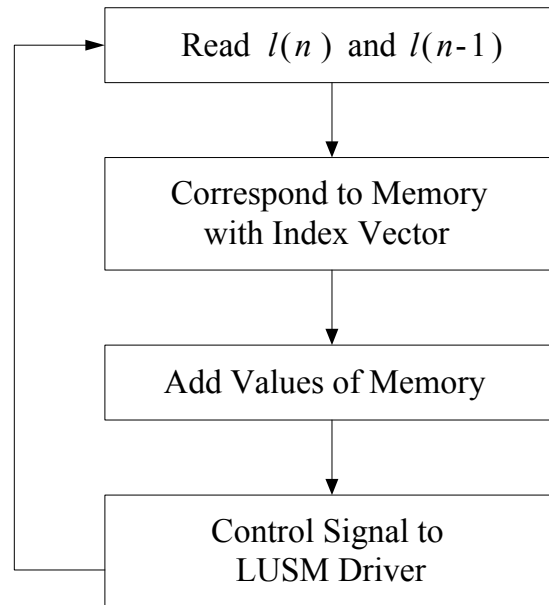


圖5-4 CMAC 控制 LUSM 的流程圖

#### 5.4 小腦模型控制器理論分析

本篇論文使用二維小腦模型控制器來控制 LUSM，並規劃 2 層量化層( $Q_1$ ,  $Q_2$ )，主要的控制方法是來學習 LUSM 空載時的數學模型，其記憶體分割方式如圖 5-5 所示，橫軸表示  $l(n)$ ，代表目前起點與終點的相對距離，縱軸表示  $l(n-1)$ ，代表上一次起點與終點的相對距離；將相對距離分成 16 等分，每一等分將會對應到 2 個實體記憶體，共有 256 個參考狀態；當  $l(n) \neq 0$  且  $l(n) = l(n-1)$ ，代表在此相對距離區域所加的負載過大，以致 LUSM 無法移動，此時的控制電壓將會增加修正量，如虛線所示。

$Q_1$ 層在橫軸分割成 8 等分(A~H)，每一等分對應到 8 個記憶體空間(a~h)，需使用 64 個記憶體空間， $Q_2$ 層在橫軸分割成 9 等分(I~Q)，每一等分對應到 9 個記憶體空間(i~q)，需使用 81 個記憶體空間，總共須使用 145 個記憶體空間。當輸入的參考狀態為(10, 3)，如表 5-2 所示，經由記憶體索引指標建立表，所對應的記憶體為(Fb)與(Nk)兩個實體記憶體，被對應到記憶體索引指標為 1，其餘皆被設為 0，以參考狀態(10,3)為例，有(Fb)與(Nk)兩個實體記憶體被對應到，將 2 個記憶體資料相加即是此參考狀態的實際輸出值。其索引指標位址計算如 5-1 式與 5-2 式，所對應各量化層記憶體名稱與記憶體位址建立如表 5-3、表 5-4 所示。

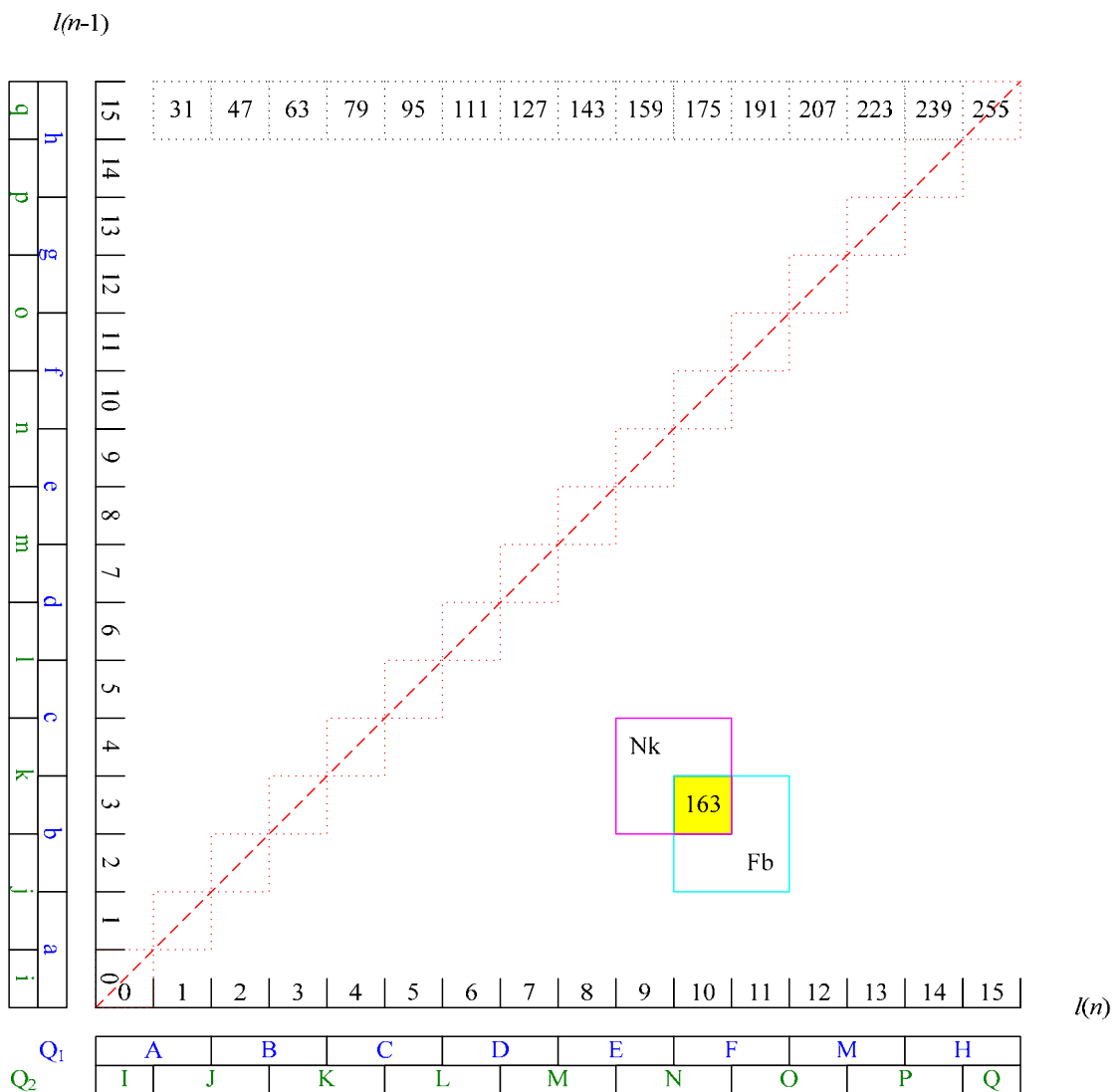


圖5-5 二維 CMAC 學習空間

表5-2 記憶體索引指標建立表

$l(n)$	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
6	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
7	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
8	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
9	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
11	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
12	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0
13	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
14	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
15	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1
	Q1 層								Q2 層								
$l(n-1)$	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
6	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
7	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
8	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
9	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
11	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
12	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0
13	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
14	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
15	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1

$$l(n)=10$$

$$l(n-1)=3$$

ceil()：為 MATLAB 軟體無條件進位的指令

$$\text{Q1層, } a(s) = 1, \quad s = \left( \text{ceil}\left(\frac{l(n)-1}{2}\right) * 8 \right) + \text{ceil}\left(\frac{l(n-1)-1}{2}\right) = 41 \quad (5-1)$$

$$\text{Q2層, } a(s) = 1, \quad s = 64 + \left( \text{ceil}\left(\frac{l(n)}{2}\right) * 9 \right) + \text{ceil}\left(\frac{l(n-1)}{2}\right) = 111 \quad (5-2)$$

表5-3 Q1 量化層記憶體名稱對應記憶體位址建立表

Q1 量化層	A	<b>Aa</b>	Ab	Ac	Ad	Ae	Af	Ag	Ah
	s	<b>0</b>	1	2	3	4	5	6	7
	B	Ba	<b>Bb</b>	Bc	Bd	Be	Bf	Bg	Bh
	s	8	<b>9</b>	10	11	12	13	14	15
	C	Ca	Cb	<b>Cc</b>	Cd	Ce	Cf	Cg	Ch
	s	16	17	<b>18</b>	19	20	21	22	23
	D	Da	Db	Dc	<b>Dd</b>	De	Df	Dg	Dh
	s	24	25	26	<b>27</b>	28	29	30	31
	E	Ea	Eb	Ec	Ed	<b>Ee</b>	Ef	Eg	Eh
	s	32	33	34	35	<b>36</b>	37	38	39
	F	Fa	<b>Fb</b>	Fc	Fd	Fe	<b>Ff</b>	Fg	Fh
	s	40	<b>41</b>	42	43	44	<b>45</b>	46	47
	G	Ga	Gb	Gc	Gd	Ge	Gf	<b>Gg</b>	Gh
	s	48	49	50	51	52	53	<b>54</b>	55
	H	Ha	Hb	Hc	Hd	He	Hf	Hg	<b>Hh</b>
	s	56	57	58	59	60	61	62	<b>63</b>



表5-4 Q2 量化層記憶體名稱對應記憶體位址建立表

Q2 量化層	I	<b>Ii</b>	Ij	Ik	Il	Im	In	Io	Ip	Iq
	s	<b>64</b>	65	66	67	68	69	70	71	72
	J	Ji	<b>Jj</b>	Jk	Jl	Jm	Jn	Jo	Jp	Jq
	s	73	<b>74</b>	75	76	77	78	79	80	81
	K	Ki	Kj	<b>Kk</b>	Kl	Km	Kn	Ko	Kp	Kq
	s	82	83	<b>84</b>	85	86	87	88	89	90
	L	Li	Lj	Lk	<b>Ll</b>	Lm	Ln	Lo	Lp	Lq
	s	91	92	93	<b>94</b>	95	96	97	98	99
	M	Mi	Mj	Mk	Ml	<b>Mm</b>	Mn	Mo	Mp	Mq
	s	100	101	102	103	<b>104</b>	105	106	107	108
	N	Ni	Nj	<b>Nk</b>	Nl	Nm	<b>Nn</b>	No	Np	Nq
	s	109	110	<b>111</b>	112	113	<b>114</b>	115	116	117
	O	Oi	Oj	Ok	Ol	Om	On	<b>Oo</b>	Op	Oq
	s	118	119	120	121	122	123	<b>124</b>	125	126
	P	Pi	Pj	Pk	Pl	Pm	Pn	Po	<b>Pp</b>	Pq
	s	127	128	129	130	131	132	133	<b>134</b>	135
	Q	Qi	Qj	Qk	Ql	Qm	Qn	Qo	Qp	<b>Qq</b>
	s	136	137	138	139	140	141	142	143	<b>144</b>

### 5.3.1 小腦模型控制器學習樣本

在規劃好 CMAC 的學習空間、量化層的數量與參考狀態的個數後，接下來要將 LUSM 的數學模型讓 CMAC 去學習，使用先前所製作 RBC 來控制超音波馬達的定位，大約繪製出 LUSM 的數學模型，如圖 5-2 所示，在空載條件下，超音波馬達的  $l(n)$  對應控制電壓並繪製成曲線圖，為了加強 CMAC 對超音波馬達定位的性能與適應能力，本篇論文將 CMAC 的學習空間擴充為二維學習空間，如表 5-5 所示，可以使 LUSM 在數學模型或負載改變時，CMAC 依然有良好的控制性能。

經過多次反覆實驗後，可以將超音波馬達的  $l(n)$  對應控制電壓，歸納出 4 點特性，最後再使用這 4 點特性作為擴充二維 CMAC 學習空間的依據，如下列各

點：

1. 當  $l(n)=0$ ，代表 LUSM 已經在終點上，CMAC 輸出的控制電壓為 0。
2. 當  $l(n)\neq 0$  且  $l(n)=l(n-1)$ ，代表 LUSM 的數學模型改變或負載過大，以致 LUSM 停在原點無法移動，CMAC 輸出的控制電壓將往上一個距離等級 ( $l(n)$ ) 修正，迫使 LUSM 移動。
3. 當  $l(n)\neq 0$ 、 $l(n)<l(n-1)$ 、 $l(n)$  不變且  $l(n-1)$  越來越小時，代表期望值的變化率大於實際值的變化率，CMAC 輸出的控制電壓將往同一距離等級 ( $l(n)$ ) 的最高點修正。
4. 當  $l(n)\neq 0$ 、 $l(n)>l(n-1)$ 、 $l(n)$  不變且  $l(n-1)$  越來越大時，代表期望值的變化率小於實際值的變化率，CMAC 輸出的控制電壓將往同一距離等級  $l(n)$  的最低點修正。

表5-5 超音波馬達的  $l(n)$  對應控制電壓表(二維空間)

$y_s$		$l(n-1)$																
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
$l(n)$	0	<b>0</b>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	1	90	<b>96</b>	89	89	88	88	88	87	87	87	86	86	86	85	85	85	
	2	110	110	<b>115</b>	109	108	108	108	107	107	107	106	106	106	105	105	105	
	3	124	124	124	<b>129</b>	123	123	123	122	122	122	121	121	121	120	120	120	
	4	139	138	138	138	<b>143</b>	137	137	136	136	136	135	135	135	134	134	134	
	5	152	152	151	151	151	<b>156</b>	150	149	149	149	148	148	148	147	147	147	
	6	163	163	163	162	162	162	<b>167</b>	161	161	161	160	160	160	159	159	159	
	7	175	174	174	174	173	173	173	<b>178</b>	172	172	171	171	171	170	170	170	
	8	185	185	184	184	184	183	183	183	<b>188</b>	182	181	181	181	180	180	180	
	9	193	193	193	192	192	192	191	191	191	<b>195</b>	190	190	190	189	189	189	
	10	202	201	201	201	200	200	200	199	199	199	<b>204</b>	198	198	197	197	197	
	11	209	209	208	208	208	207	207	207	206	206	206	<b>211</b>	205	204	204	204	
	12	214	214	214	213	213	213	212	212	212	211	211	211	<b>216</b>	210	210	210	
	13	220	219	219	219	218	218	218	217	217	217	216	216	216	<b>221</b>	215	215	
	14	224	224	223	223	223	222	222	222	222	221	221	221	220	220	220	<b>225</b>	219
	15	230	230	230	229	229	229	228	228	228	227	227	227	226	226	226	226	<b>232</b>

### 5.3.2 小腦模型控制器記憶體權值

使用 MATLAB 軟體在個人電腦上進行模擬，將表 5-5 的內容( $y_s$ )代入 5-3 式、5-4 式、5-5 式中，求出每一個真實記憶體的權重( $w$ )，經過多次模擬測試後，將學習率  $\alpha$  設定在 0.41，可以有最佳的學習率；如圖 5-6 所示，為 CMAC 的期望值輸出，在經過 100 次的學習訓練後，產生 CMAC 的實際值輸出，如圖 5-7 所示，比較兩圖的結果，可以證明使用 CMAC 可以有效學習非線性的數學模型，再將兩圖的輸出值相減求絕對值，繪製出學習次數對應絕對值誤差總和曲線圖，如圖 5-8 所示，可以證明使用 CMAC 具有快速學習約收斂的特性，當迭代至 44 次時，誤差值已經降至最小微幅震盪，如圖 5-9 所示。

$w$ : 權重

$y_s$ : 真實輸出

$\alpha$ : 學習率(0.41)

$j$ : 真實記憶體編號

$i$ : 目前學習的週期數

$s$ : 目前學習的樣本編號

$\bar{y}_s$ : 第 $s$ 樣本的實際值輸出

$N_e$ : 每一個樣本所對應的真實記憶體數量(層數=2)

$a$ : 真實記憶體索引指標(聯想向量 association vector)

$$\bar{y}_s = [a_{s,1}, a_{s,2}, \dots, a_{s,N_h}] \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_{N_h} \end{bmatrix} = \sum_{j=1}^{N_h} a_{s,j} w_j = a_s^T w \quad (5-3)$$

$$w_s^{(i)} = w_{s-1}^{(i)} + \Delta w_{s-1}^{(i)} \quad (5-4)$$

$$\Delta w_{s-1}^{(i)} = \frac{\alpha}{N_e} a_{s-1} (\bar{y}_{s-1} - a_{s-1}^T w_{s-1}^{(i)}) \quad (5-5)$$

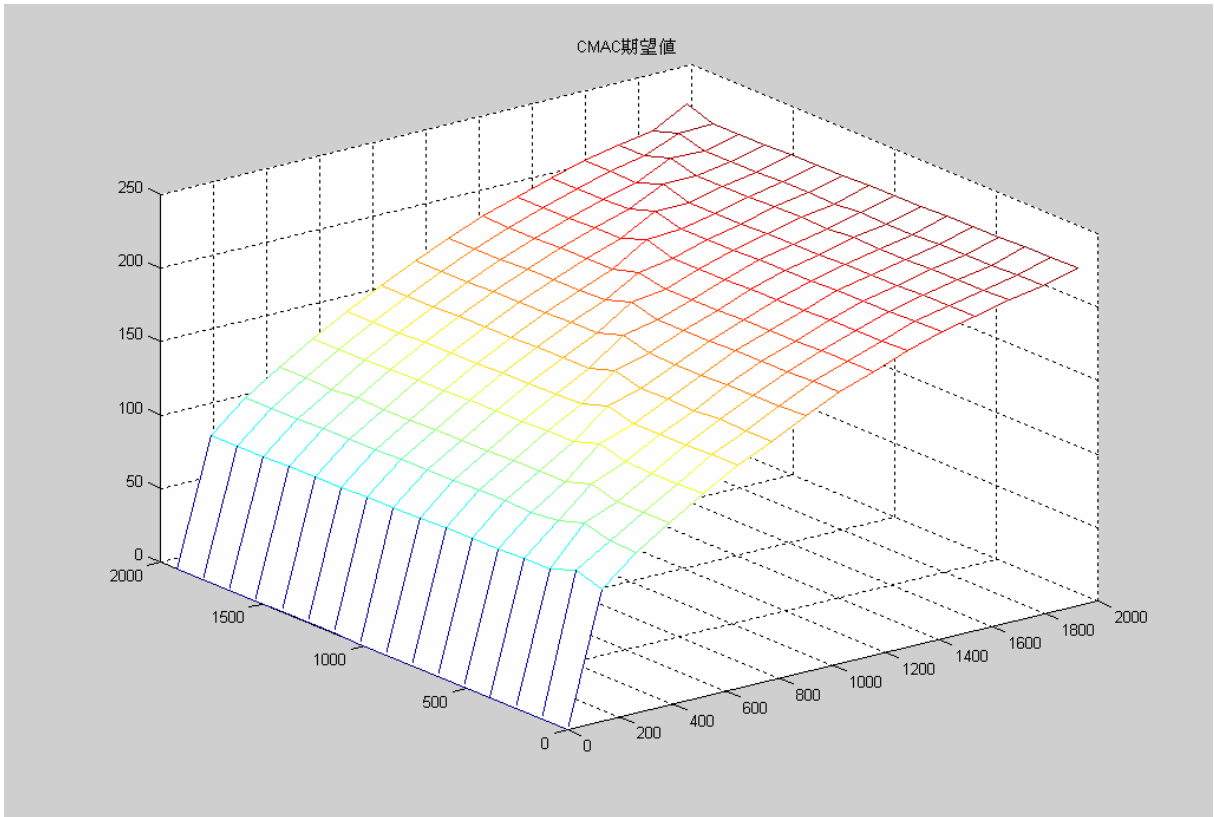


圖5-6 二維 CMAC 學習樣本(期望值)

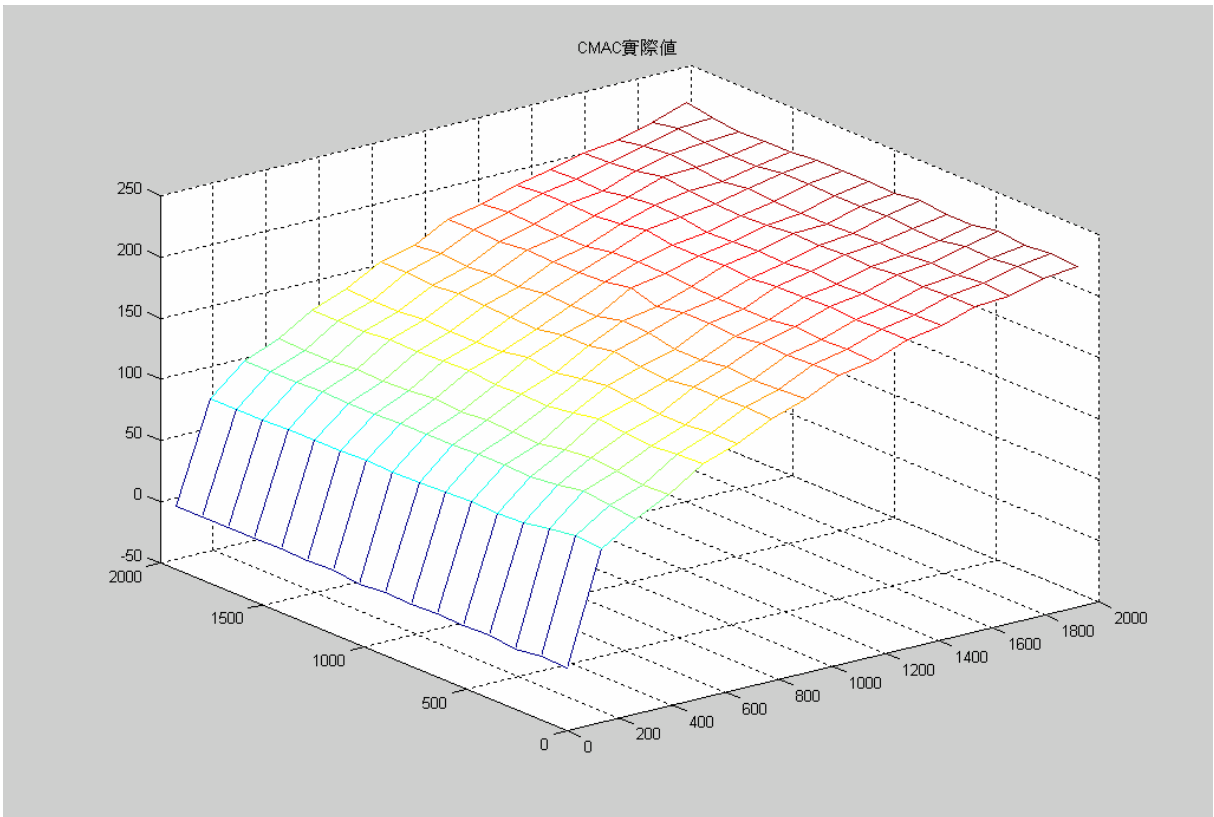


圖5-7 二維 CMAC 學習結果(實際值)

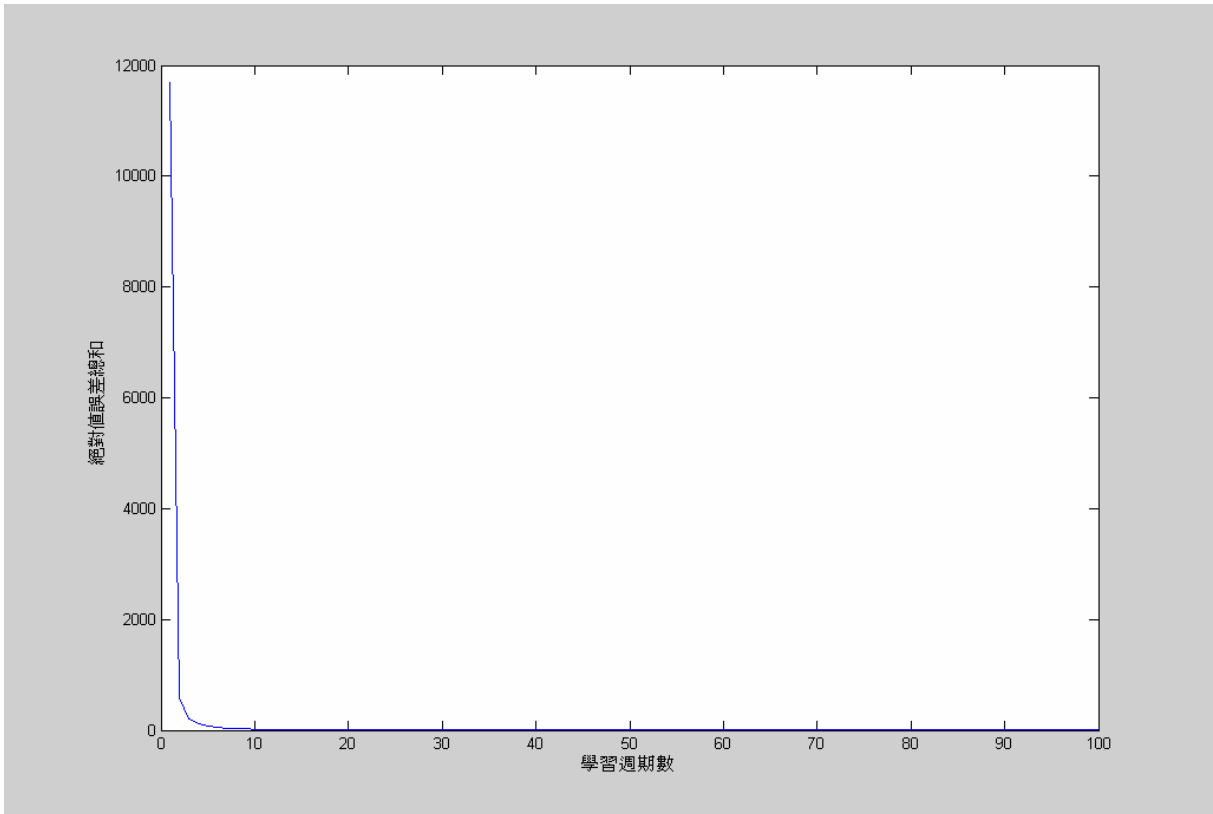


圖5-8 學習次數對應絕對值誤差總和曲線圖

Array Editor: SE																		
File Edit View Web Window Help																		
Numeric format: shortG Size: 1 by 100																		
	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
1	3.0391	3.0586	2.9531	2.8945	2.9102	2.8789	2.8438	2.8438	2.793	2.7539	2.7461	2.6328	2.6328	2.6328	2.6328	2.6328	2.6328	2.6328

圖5-9 學習次數對應絕對值誤差總和表

最後，將每一個實體記憶體的權值寫入至 PIC18F452 的程式記憶體中，當有期望值信號輸入時，會將此信號(類比信號)轉換成數位信號對應一個參考狀態，每一個參考狀態對應一個索引向量，每一個索引向量對應一組實體記憶體，使用程式記憶體索引指令(tblrd)可以很容易的將所需實體記憶體中資料取出，並將此資料相加，即是控制電壓的輸出，記憶體名稱、位址與權重如表 5-6、表 5-7 所示。

表5-6 Q1 量化層記憶體位址對應權重表

Q1 層, $a(s), s=(\text{ceil}((e(n)-1)/2)) * 8 + \text{ceil}((e(n-1)-1)/2)$	A	<b>Aa</b>	Ab	Ac	Ad	Ae	Af	Ag	Ah
	s	<b>0</b>	1	2	3	4	5	6	7
	$w_s$	<b>35</b>	32	31	30	29	28	28	28
	B	Ba	<b>Bb</b>	Bc	Bd	Be	Bf	Bg	Bh
	s	8	<b>9</b>	10	11	12	13	14	15
	$w_s$	53	<b>56</b>	54	54	53	51	52	52
	C	Ca	Cb	<b>Cc</b>	Cd	Ce	Cf	Cg	Ch
	s	16	17	<b>18</b>	19	20	21	22	23
	$w_s$	71	69	<b>72</b>	69	69	68	69	69
	D	Da	Db	Dc	<b>Dd</b>	De	Df	Dg	Dh
	s	24	25	26	<b>27</b>	28	29	30	31
	$w_s$	85	84	81	<b>85</b>	81	82	83	85
	E	Ea	Eb	Ec	Ed	<b>Ee</b>	Ef	Eg	Eh
	s	32	33	34	35	<b>36</b>	37	38	39
	$w_s$	107	106	103	103	<b>106</b>	105	105	106
	F	Fa	<b>Fb</b>	Fc	Fd	Fe	<b>Ff</b>	Fg	Fh
	s	40	<b>41</b>	42	43	44	<b>45</b>	46	47
	$w_s$	117	<b>116</b>	115	113	113	<b>117</b>	115	115
	G	Ga	Gb	Gc	Gd	Ge	Gf	<b>Gg</b>	Gh
	s	48	49	50	51	52	53	<b>54</b>	55
	$w_s$	126	124	122	121	120	121	<b>124</b>	122
	H	Ha	Hb	Hc	Hd	He	Hf	Hg	<b>Hh</b>
	s	56	57	58	59	60	61	62	<b>63</b>
	$w_s$	133	131	127	129	127	127	127	<b>130</b>

表5-7 Q2 量化層記憶體位址對應權重表

Q2, a(s), s=64+(ceil(e(n)/2))*9+ceil(e(n-1)/2	I	<b>Ii</b>	Ij	Ik	Il	Im	In	Io	Ip	Iq
	s	<b>64</b>	65	66	67	68	69	70	71	72
	w <sub>s</sub>	<b>-33</b>	-33	-30	-29	-28	-27	-26	-26	-26
	J	Ji	<b>Jj</b>	Jk	Jl	Jm	Jn	Jo	Jp	Jq
	s	73	<b>74</b>	75	76	77	78	79	80	81
	w <sub>s</sub>	57	<b>59</b>	55	56	55	56	56	55	55
	K	Ki	Kj	<b>Kk</b>	Kl	Km	Kn	Ko	Kp	Kq
	s	82	83	<b>84</b>	85	86	87	88	89	90
	w <sub>s</sub>	70	69	<b>71</b>	67	68	69	68	66	66
	L	Li	Lj	Lk	<b>Ll</b>	Lm	Ln	Lo	Lp	Lq
	s	91	92	93	<b>94</b>	95	96	97	98	99
	w <sub>s</sub>	80	80	80	<b>82</b>	78	78	78	76	76
	M	Mi	Mj	Mk	Ml	<b>Mm</b>	Mn	Mo	Mp	Mq
	s	100	101	102	103	<b>104</b>	105	106	107	108
	w <sub>s</sub>	89	88	90	90	<b>91</b>	90	87	85	83
	N	Ni	Nj	Nk	Nl	Nm	<b>Nn</b>	No	Np	Nq
	s	109	110	111	112	113	<b>114</b>	115	116	117
	w <sub>s</sub>	85	85	87	87	88	<b>87</b>	83	84	82
	O	Oi	Oj	Ok	Ol	Om	On	<b>Oo</b>	Op	Oq
	s	118	119	120	121	122	123	<b>124</b>	125	126
w <sub>s</sub>	90	90	91	92	93	91	<b>92</b>	88	87	
P	Pi	Pj	Pk	Pl	Pm	Pn	Po	<b>Pp</b>	Pq	
s	127	128	129	130	131	132	133	<b>134</b>	135	
w <sub>s</sub>	92	93	94	95	94	96	93	<b>95</b>	91	
Q	Qi	Qj	Qk	Ql	Qm	Qn	Qo	Qp	<b>Qq</b>	
s	136	137	138	139	140	141	142	143	<b>144</b>	
w <sub>s</sub>	95	97	100	100	97	100	98	97	<b>100</b>	