العودة إلى البرنامج المستدعي عن طريق إعادة تخزين قيم IP فقط أو IP و CS معاً (حسب نوع تعليمة المناداة أي ضمن المقطع الجزئي أو خارجه) من أجل المتحول Far_pro . و إذا كان المتحول (operand) موجوداً في تعليمة العودة RET فيجب إضافته إلى محتويات SP . هذا و إن المتحول إذا وجد في تعليمة العودة فهو عبارة عن متحول إزاحة بـ 16 بت.

تاسعاً – تعليمات الدفع و السحب

إن التعليمة المستخدمة لحفظ البارامترات في المكدس هي تعليمة الدفع PUSH و التعليمة المستخدمة لاسترجاعها هي تعليمة POP . بعد سياق التحويل إلى البرنامج الفرعي نجد أنه من الضروري عادة حفظ محتويات المسجلات الرئيسية أو بعض بارامترات البرنامج الرئيسي هذه القيم يتم حفظها بواسطة دفعها إلى المكدس. و بهذه الطريقة يتم حفظ المحتويات سليمة في مقطع المكدس للذاكرة أثناء تنفيذ البرنامج الفرعي، و قبل العودة إلى البرنامج الرئيسي فإن المسجلات المحفوظة و بارامترات البرنامج الرئيسي يُعاد تخزينها بواسطة سحب القيم المحفوظة من المكدس. لذلك فإن البنية النموذجية للبرنامج الفرعي تكون كالتالي:

```
PUSH xx
PUSH yy        ← لحفظ المسجلات و البارامترات في المكدس
PUSH zz
----------
----------     ← جسم البرنامج الفرعي
----------
POP zz
POP yy         ← لاسترجاع المسجلات و البارامترات من المكدس
POP xx
RET            ← للعودة إلى البرنامج المستدعي
```

نلاحظ أنه يتم الاسترجاع هنا بشكل عكسي و ذلك لأن مبدأ عمل المكدس هو LIFO

ملاحظة: يتعامل المكدس مع كلمات و ليس مع بايتات.

تعليمات PUSH, POP

و هي مشروحة في الجدول التالي:

الأعلام المتأثرة	العملية	الصيغة	المعنى	الكلمة المختزلة
لا يوجد	S → ((SP))	PUSH S	دفع كلمة إلى المكدس	PUSH
لا يوجد	((SP)) → D	POP D	سحب كلمة من المكدس	POP

المكدس، مسجل مقطع المكدس SS ،مؤشر المكدس SP

أثناء عمليات المقاطعة ومناداة البرنامج الفرعي يتم دفع محتويات المسجلات الداخلية المعّينة بالمعالج إلى قسم من الذاكرة يدعى بالمكدس حيث تبقى هذه المحتويات هناك بشكل مؤقت. وعند إكمال روتين خدمة المقاطعة أو البرنامج الفرعي يتم سحب هذه القيم من المكدس وتوضع في نفس المسجل الداخلي حيث كان يحتوها أصلاً. فمثلاً عندما تحدث المقاطعة فإن المعالج و بشكل